MODELING THE EVOLUTION OF CORROSION: A FEATURE-BASED MODEL FOR GROWTH

Presented to

the Faculty of the School of Engineering and Applied Science

University of Virginia

Department of Systems and Information Engineering

In Partial Fulfillment

of the Requirements for the Degree

Doctor of Philosophy in Systems Engineering

by

Major Brian E. Ralston

August 2006

The views expressed in this dissertation are those of the author and do not reflect the official policy or position of the United States Air Force, Department of Defense, or the U.S. Government.

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE **01 AUG 2006**	2. REPORT TYPE **N/A**	3. DATES COVERED **-**
4. TITLE AND SUBTITLE **Modeling The Evolution Of Corrosion: A Feature-Based Model For Growth**		5a. CONTRACT NUMBER 5b. GRANT NUMBER 5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)		5d. PROJECT NUMBER 5e. TASK NUMBER 5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **University of Virginia**		8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) **AFIT/CIA**		10. SPONSOR/MONITOR’S ACRONYM(S) 11. SPONSOR/MONITOR’S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release, distribution unlimited

13. SUPPLEMENTARY NOTES

, The original document contains color images.

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**UU**	**178**	

Brian Ralston

August 2006

APPROVAL SHEET

This dissertation is submitted in partial fulfillment of the requirements for the degree of
Doctor of Philosophy in Systems and Information Engineering

Major Brian E. Ralston (USAF), Author

This dissertation has been read and approved by the Examining Committee:

Stephen D. Patek, Committee Chair
Department of Systems and Information Engineering

Donald E. Brown, Dissertation Advisor
Department of Systems and Information Engineering

Robert G. Kelly
Department of Materials Science and Engineering

Michael Devore
Department of Systems and Information Engineering

Ginger M. Davis
Department of Systems and Information Engineering

Accepted for the School of Engineering and Applied Science:

Dean, School of Engineering and Applied Science

August 2006

TABLE OF CONTENTS

ABSTRACT

The Government Accounting Office reports that the Department of Defense spends approximately $20 Billion per year on prevention and repair of corrosion. The resources required to combat corrosion problems are desperately needed for equipment and personnel in today's high operations tempo environment. Despite its significance, modeling for the prediction of corrosion evolution has been largely ignored in the literature.

Corrosion is the result of electrochemical reactions that initiate at a given site on the metal, and continue to propagate as long as the environmental conditions can support the reaction. The process of corrosion can be decomposed into three phases: initiation, growth, and death. Although some models have been developed which capture these phases (Macdonald and Engelhardt, 2003), most models for corrosion are developed at a laboratory scale utilizing variables only measurable in a highly controlled and instrumented environment. In addition, most attempts to model corrosion disregard the spatial nature of the phenomena.

The initiation of corrosion at specific locations can be represented by a space-time shock point process. Unfortunately, traditional point process models only capture the relationship in time and space. Each geographic location maps to a point in feature space which contains information that defines that location (e.g. environmental conditions, distance to other corroded sites). The incorporation of feature space into a point process model has been shown to improve the model's predictive ability (Liu and Brown, 2003). Similarly, spatial models for growth do not capitalize on the ability of features associated with spatial locations to predict growth. Growth models, such as random closed sets and interacting particle models represent growth by placing area patterns around an initial configuration. With typical interacting particle models, the local transition function is assumed known so that the relationship between the spatial locations and environmental features is understood.

In this work, a feature-based model for event initiation and growth in space and time is developed. The initiation model is based on previous research for feature-based prediction of event locations. The growth model expands the basic interacting particle system definition to incorporate feature information using a growth probability function defined over the geographic region of interest. The inclusion of significant features improves the approximation of the function in terms of error and deviance reduction.

The model is applied using data from images of filiform growth in samples of AA2024-T3. The first approach involves the derivation of features from macro scale images where the only distinguishable features are geometric in nature and derived from the filiform growth objects. Segmentation of individual filaments is not possible at this scale and a wide range of classification methods are considered to predict filiform growth using the derived feature information. The results show that the derived features at this scale are insufficient to capture the directed nature of the filiform growth process.

In order to improve upon the previous application, micro scale data were generated so that heterogeneities in the material can be incorporated into the feature space. The heterogeneities are classified by intensity level, and the density of each class of heterogeneity is used as the feature set at each geographic location. At this scale, initiation and growth of individual filaments can be captured. The heterogeneity features are used to approximate the spatial transition density in feature space using kernel density estimates. The results from the growth model are generated through simulation of the proposed interacting particle model. The growth probability function for the model is estimated using logistic regression, classification tree and neural network methods.

Finally, the evaluation of the feature-based models verifies the theoretical assumption that the inclusion of the feature information into the initiation and growth models improves the predictive ability when compared to a spatially random baseline.

For initiation, results show that in 8 of the 9 areas considered, the initiation locations are more likely generated from the feature-based transition density in comparison to a random spatial density. Likewise, the results for the growth model with a 2nd order logistic regression growth probability function offer significant improvement over a random spatial growth model for filaments in the training and test samples, as well as filaments in a separate validation sample.

ACKNOWLEDGEMENTS

First and foremost, I would like to thank my wife for the love and support that she has provided me throughout our years together. She has sacrificed a great deal as I have progressed from assignment to assignment, and certainly these three years have been no different. She has always been there when my confidence was shaken to pick me up and encourage me, and to motivate me on those days when I didn't want to continue. I can never thank you enough for all you do for me and our family. I love you.

Next I would like to thank my three sons, for everyday helping me to realize the truly important aspects of life. I am so proud of each of you and I am thankful to have had the flexibility over the last three years to be with you at times when I otherwise would not have been able. You too bear the burden of being moved around from assignment to assignment, and for that I am sorry, but I hope that in some ways at least there are experiences to be gained from it. I love each of you and as sad as it is, I look forward to watching you grow to young men.

I would also like to thank my committee members for their guidance and insight throughout this process. I have learned a great deal from working with you on this research and from the courses I have taken from you. Specifically, I would like to thank Prof Rob Kelly, who has taught me more about corrosion than I ever wanted to know! However, he has guided and advised me on most every aspect of this research, and I am certainly grateful to have worked with him. I would also like to thank my advisor, Prof Don Brown for all of his direction and vigilance throughout this research and my time in the Systems department. It has been a long and arduous process, but a rewarding one and something I have truly enjoyed.

The financial support of this work by the Office of Naval Research (Dr. A. Perez), Contract N00014-05-1-0251 is gratefully acknowledged. In addition, I want to thank Connor Parker for all of the work he has done to generate the samples used in this research. I look forward to working with you further, and I wish you the best of luck in completing your research!

Finally, I need to thank the guys that carried me through most of this: Scott, Mike, Donald, and He-man. You guys were always there to listen to my poorly explained ideas and look at the countless images of corrosion growth – and you usually pretended you were interested! Seriously, thank you for making this a great experience and being great friends!!

LIST OF FIGURES

LIST OF TABLES

LIST OF APPENDICES

CHAPTER 1. INTRODUCTION

The first chapter introduces the background and motivation that prompted the research in corrosion prediction. The objectives and contributions of this dissertation are given as well as an outline for subsequent chapters.

1.1. Background and Motivation

The Air Force Research Laboratory (AFRL, 2004b), reports that one of the most formidable enemies facing the U.S. Air Force today is corrosion attacking the aging fleet of aircraft. In FY 2007, the average age of the U.S. Air Force fleet will be over 25 years, and while the Air Force continues to stretch the life of each aircraft, the stress and strain on airframes builds (Garamone, 2001). A large percentage of commercial and military aircraft in operation today, have met or exceeded their original design life, which dramatically increases the required maintenance and repair costs (NAP, 1997). As the acquisition of new equipment decreases and the age of current systems increases, corrosion is potentially the most significant driver of life-cycle cost (Virmani, 2002). The annual cost of corrosion to the U.S. commercial airline industry is over $2.0 Billion: $1.7 Billion for corrosion-related maintenance and $0.3 Billion for corrosion-related downtime (Virmani, 2002). Corrosion prevention and maintenance is currently inadequate primarily because of the lack of understanding of the corrosion process, specifically the inability to predict the initiation and growth of corrosion in airframe components (Virmani, 2002).

The effect of corrosion on aging aircraft is particularly applicable to the fleet of Air Force tankers. The KC-135 is the oldest aircraft in the Air Force inventory, with an average age of over 40 years. The Air Force maintains a fleet of over 500 KC-135's, and with no immediate plans for replacement, the aircraft could remain in service for another 25 years (NAP, 1997). The extended life of the KC-135 and other Air Force aircraft translates into an increase in fatigue, wear, and corrosion damage. Corrosion decreases the overall strength of the airframe due to the loss in skin thickness, fatigue crack initiation, and increased crack growth rates (Koch *et al.*, 1998). The problems caused by corrosion have a direct impact on the maintenance costs and operational readiness of the fleet due to the increased time required for depot level repair (NAP, 1997). In today's high operational tempo, the Air Force relies heavily on the tanker fleet to perform missions around the world. Each year, the Air Force spends between $1-3 Billion on operations related to corrosion. The money spent on the prevention and repair of corrosion, is desperately needed for equipment and system upgrades. The situation highlights the need for accurate corrosion prediction models which can be used to develop a cost effective corrosion integrity program (Koch *et al.*, 1998).

Methods for identifying corrosion range from very expensive, yet reliable, tear-down to cheaper and less reliable non-destructive inspection (NDI). One of the most common NDI methods is eddy current scanning. Eddy current scans are performed by an operator passing a coil over the skin of the aircraft. The coil emits an alternating current which causes fluctuations in the magnetic fields on the surface. Defects under the skin disrupt the flow of current and result in changes to voltage readings. Voltage data is then translated into a false color image (Figure 1) which is visually interpreted by the operator. The interpretation of the image is extremely susceptible to bias and inconsistency (Brence and Brown, 2002b).

Currently, corrosion is not included in a damage tolerance estimate, so when corrosion is found, the policy is to immediately repair (Koch *et al.*, 1998). The Air Force developed the structural damage management tool (SDMT), a software application developed to produce predictions for crack growth based on the current corrosion image provided by eddy current scan. SDMT uses the image to create a distribution for the amount of material loss due to corrosion along a line of pixels through the suspected crack. Figure 2 shows an example screen shot from SDMT. The current and predicted distribution for material loss along a selected line of pixels in the given image is shown.

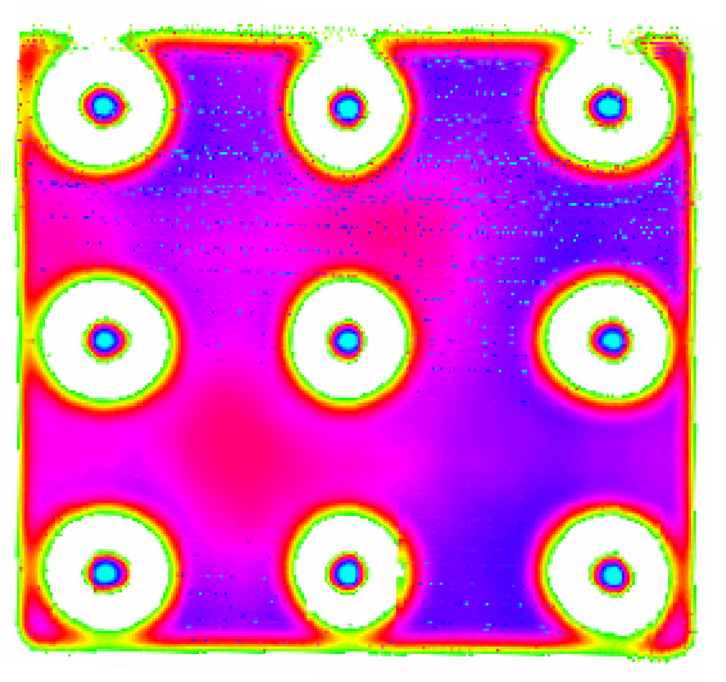
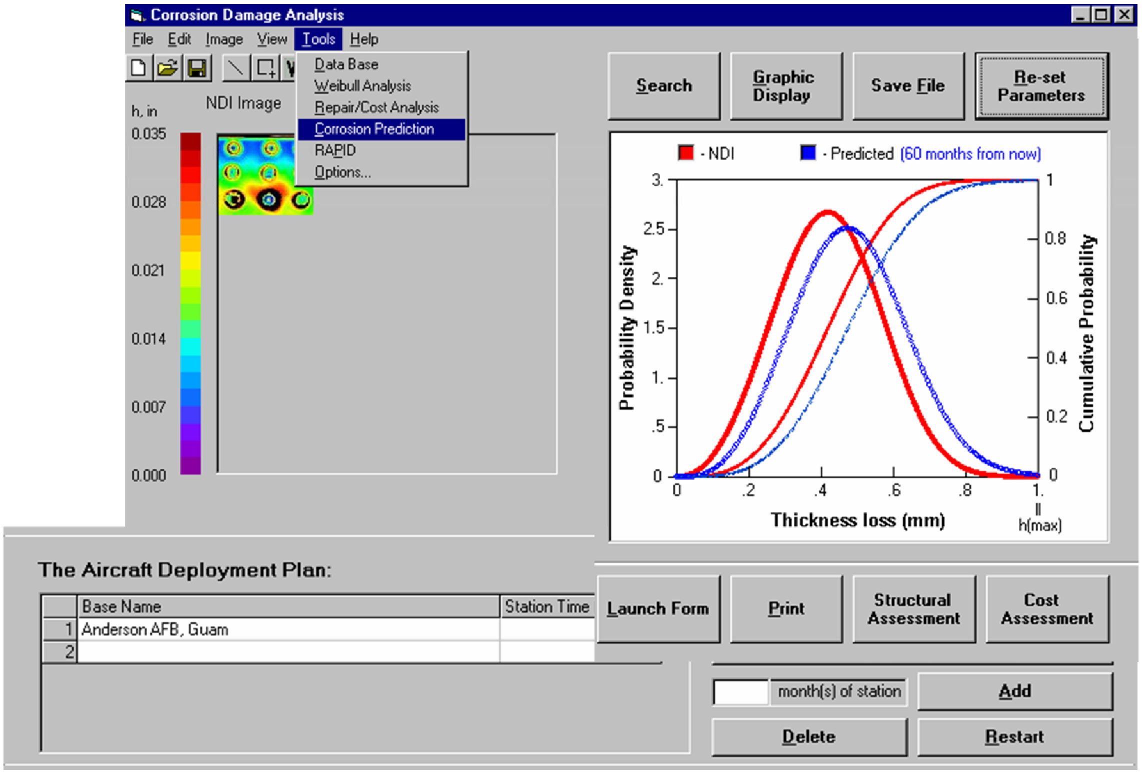


Figure 1. Visual representation of eddy current response



The Aircraft Deployment Plan:

	Base Name	Station Time
1	Anderson AFB, Guam	
2		

Figure 2. Screenshot of SDMT

SDMT accounts for the basing strategy of the aircraft by allowing the user to input the projected location and length of time that the aircraft will be based at each location. The basing strategy is used to determine the rates of corrosion that are needed to predict crack growth. Given the current distribution and projected basing strategy, SDMT predicts the future distribution of material loss along the line of pixels. The mean value of this distribution is then supplied as input to a fracture mechanics application, AFGROW, a crack growth analysis tool designed to predict the life of a material once a crack has formed (AFRL, 2004a).

The basis for current corrosion prediction methods revolves around the need to determine the rate of corrosion for a metal given certain environmental conditions. In SDMT, the location specific corrosion rates are derived from data obtained from experiments with sample coupons of a similar metal exposed at 7 bases throughout the world (Kadena AFB, MacDill AFB, Hickam AFB, Tinker AFB, Pease ANGB, Mildenhall AFB, and Seymour-Johnson AFB). The rates for other metals and locations are then extrapolated from the data gained from these experiments. SDMT uses the rates to predict the future distribution of material loss given the proposed basing strategy.

Despite the spatial nature of the data in the NDI images, all spatial information is discarded when material loss along a single line is considered. SDMT is only concerned with a line of pixels because of the stress loads that affect how the crack propagates. This methodology corresponds to an assumption that the corrosion rate, and thus the predicted amount of material loss, at a given pixel is independent of the corrosion that exists at other pixels.

1.2. Objective and Contributions

This dissertation seeks to address space-time prediction of initiation and growth for random events. The overall objective is to build on the theory of stochastic point processes and growth models to develop an approach that utilizes the features significant for initiation and growth of a random system and provides a probabilistic prediction for future occurrence. The application of the proposed methodology is for the prediction of corrosion of metals, but the model is applicable to similar phenomena, such as growth of biological organisms, fire spread, or urban sprawl. There are 4 primary contribution areas for this research:

- The fundamental goal of this research is to formulate a spatiotemporal model that incorporates feature space analysis in space-time event initiation and growth prediction. By extending the previously developed feature-based point prediction model (Liu, 1999) to a model for growth prediction, we can produce a probability map to drive the behavior of an interacting particle model which can be used to predict growth over a given spatial region.
- The application of the model is demonstrated with data of filiform corrosion growth in aluminum. Images of filiform growth on samples of AA2024-T3 are used to produce data to fit the model components. The features considered are determined by the scale of the data images.
- The ability of the model to predict the growth of filiform corrosion compared to a hypothesis of random spatial growth is evaluated. Due to the lack of other predictive models, a model of random growth is used as a baseline for comparison. Simulations provide numerous realizations of predicted growth for the feature-based and random models.
- The effectiveness of the proposed model in terms of error reduction is shown and verified using several metrics including total accuracy, precision, recall, and F-measure.

1.3. Remainder of Dissertation

The remainder of this dissertation describes the various models and methodologies developed to analyze and predict corrosion growth, as well as the results and conclusions produced from the application to actual corroded samples. Chapter 2 is a literature review of four main areas of research that the present work draws upon: corrosion fundamentals, stochastic point processes, classification models, and growth processes. The goal of the review is to introduce the current state of corrosion modeling and highlight the methods used in similar problems which are applicable to this research.

The third chapter describes the methodology used to predict the location and time of corrosion initiation and growth. The formal definitions and notation are specified for both the initiation and growth models. Next, a summary of the data processing of time-sequenced images used in the application of this methodology is given. Finally, the chapter provides more detail for the classification models considered.

The theoretic properties of the model and the metrics used for evaluation are discussed in Chapter 4. It is shown that the inclusion of the feature variables into the transition functions for initiation or growth results in lower sum of squared error or deviance in the function approximation. The classification evaluation metrics for precision, recall, and f-measure are also explained.

Chapter 5 describes the two scales of data used in the application of the proposed models. First, the macro scale approach is described including the available data and derived features, followed by a description of the micro scale approach. Further detail is provided for the data and features used at the micro scale due to the significance of the inclusions as features for the model.

Chapters 6 and 7 provide the results of the models at the macro and micro scales. The macro scale model results in Chapter 6 include a test for the ability of the derived features to predict growth from a selected time period. The resolution of the data, and the preliminary results did not warrant a complete evaluation initiation and growth models. The higher resolution data in the micro scale approach allowed for the use of the initiation and interacting particle growth model.

The final chapter summarizes the conclusions from the research. Suggestions for future work are given to further refine the modeling of filiform corrosion.

CHAPTER 2. RELATED RESEARCH

This chapter serves as a literature review of four main areas of research that the present work draws upon: corrosion fundamentals, stochastic point processes, classification models, and growth processes. The goal of the review is to introduce the current state of corrosion modeling and highlight the methods used in similar problems which are applicable to this research.

2.1. Corrosion Fundamentals

Corrosion is the deterioration of metal caused by interaction with its environment. Except for gold, all refined metals are inherently unstable in the natural environment. These materials are processed from minerals or ores, and have a tendency to revert back to their more stable form. Metals corrode during an electrochemical reaction where the metal gives up electrons in the presence of water and oxygen to produce a corrosion product. Figure 3 illustrates the four requirements for corrosion (Kelly *et al.*, 2003). The anode gives up electrons and corrodes, the cathode receives electrons and a corrosion product is formed, the electrolytic path (typically water) allows for ionic conduction between the two sites, and the electrical path allows for electron conduction between the sites.

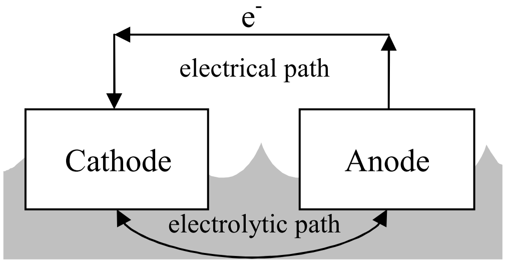


Figure 3. Requirements for corrosion

The aluminum alloys used in the KC-135 have a strong resistance to corrosion because of an oxide skin that forms when the material reacts with the atmosphere. This skin provides protection from most chemicals, weathering conditions, and even many acids; however, alkaline substances are known to penetrate the protective skin and corrode the metal (EFUNDA, 2005). The protective skin is composed of two parts: a thin, but less permeable, inner barrier layer and a much thicker outer layer. The corrosion resistance of aluminum depends on this protective skin, which is actually a hydrated form of aluminum oxide. The oxide film is naturally self-renewing and accidental abrasion or other mechanical damage of the surface film is rapidly repaired. Therefore, corrosion of aluminum occurs when the skin is abraded or when the environmental conditions cause the skin to be degraded and prevents oxygen from rebuilding. The acidity or alkalinity of the environment has a significant effect on the evolution of corrosion with aluminum alloys. Typically the corrosion will occur at low and high pH values. In high pH environments, the oxide skin is attacked more rapidly than the aluminum itself, which causes a more uniform corrosion over the surface. In acid environments, if the oxide skin is perforated locally, an accelerated attack occurs in the aluminum, which results in a pitting form of corrosion (Kingston, 1999).

2.1.1. Corrosion Analysis and Prediction Methods

Corrosion of materials is of great interest to researchers due to the criticality of the structures involved and the catastrophic result of failure. A great deal of work has been done in both the analysis and prediction of various corrosion phenomena. The study of uniform, or general, corrosion requires fewer parameters and thus analysis and modeling efforts are better established than for localized corrosion (Salvago *et al.*, 2002). Most studies of corrosion can be classified as either analysis of data to test a given hypothesis or predictive modeling to forecast evolution.

Sound statistical analysis of corrosion data can add a great deal of confidence in the results, yet many corrosion professionals are apprehensive because of the difficulty of some statistical concepts (Shibata, 2000). Despite this reluctance, a great deal of analysis has been done to reduce the distribution of corrosion data to several probability distributions. Table 1 summarizes the probability distributions observed in corrosion (Shibata, 2000).

Corrosion Data	Probability Distribution
Pitting Potential	Normal (Shibata and Takeyama, 1977)
Stress Corrosion Crack Failure Times	Log-Normal (Booth and Tucker, 1965)
2-Dimensional Distribution of Pits	Poisson (Mears and Evans, 1937)
Maximum Pit Depth	Gumbel (Aziz, 1956)
Fatigue Crack Depth	Generalized Extreme Value (Buxton and Scarf, 1994)

Table 1. Published probability distributions for corrosion

Both deterministic and probabilistic models for corrosion phenomena have been suggested. Deterministic methods account for the evolution of corrosion based on the properties constrained by the natural laws. Macdonald and Engelhardt formulate lengthy mathematical derivations of functions which explain the physical nature of pit birth, growth, and death (Macdonald and Engelhardt, 2003). These functions are then combined to describe the damage function for a localized corrosion process. The damage functions are the basis for modeling and prediction efforts, which can be adjusted based on given assumptions for pit birth and death properties. The birth rate is a function of the physical properties and time (Engelhardt *et al.*, 1999), the growth rate is assumed constant, and the death rate is a first-order decay function. Macdonald and Engelhardt show that the deterministic model can produce accurate predictions of pit penetration after one year using data for shorter periods to adjust the model parameters. Further experimentation can improve knowledge about the physical process and thus update the model and improve predictive results. The model does not account for any spatial dependence of corrosion events and therefore only provides extreme value statistics instead of a complete picture of corrosion. In fact, most of the literature covering growth and evolution deals primarily with one dimensional summary statistics rather than the complete geometric problem (Cressie and Laslett, 1987). Another drawback to this method and other deterministic models is that the data required are available only in laboratory experiments (corrosion potential, pH, passivation current, etc.).

Other authors have suggested probabilistic models for corrosion because of the inability of experimentally evaluated parameters to adequately describe the corrosion mechanism. Salvago *et al.* apply a probabilistic model for localized corrosion on a single test coupon in an aggregate model to predict general corrosion growth (Salvago *et al.*, 2002). Although very little background is given on the derivation, the local corrosion model is based on Faraday's law to provide the increase in pit depth of a local corrosion event over time periods. The aggregation of the model is justified through experimental observation of a stainless steel coupon. Based on these findings and information gained through experiments with localized corrosion, a Monte Carlo simulation is developed which models localized corrosion on many coupons as general corrosion through aggregation into one surface. The model is shown to follow experimental results for general corrosion in terms of a metric for surface roughness. This approach also assumes no spatial interaction occurs between pit events and the initiation process is homogeneous across the surface. Additionally, all spatial information produced in the model is lost because of the use of the roughness metric. A more accurate validation of the model would be to compare the predicted surface to experimental results.

2.1.2. Spatial and Temporal Characteristics

Although most corrosion modeling and analysis efforts do not address it, spatial and temporal dependence among corrosion events has been verified. Lunt, *et al.* hypothesized that the arrivals of pit events was non-Poissonian in that the arrival of a pit event is dependent on the previous arrivals (Lunt *et al.*, 1997). Experimental data verified correlation between the arrivals of pit events, especially for high intensity periods.

Budiansky and Scully set up experiments to test for spatial interaction among corrosion events (Budiansky and Scully, 2002). The birth of a pit on a material causes the environment in the surrounding area to change, which can alter the susceptibility of the surface to further pit development (Budiansky and Scully, 2002). The physics evolve to the measurement of pitting potential as a means of designing the experiment to test the hypothesis. Because the spatial reference is important, the authors use a surface consisting of a bundle of electrically isolated wires for experimental control and measurement fidelity at specific locations. The change in pitting potential with distance from an active pit increased. This result is used to support the hypothesis. The results suggest that regions adjacent to active pits are suppressed from further pitting, while areas outside the suppressed region are more susceptible to pitting (Budiansky and Scully, 2002).

2.2. Applicable Modeling Techniques

There are three phases of evolution for corrosion events: initiation, growth, and death. Macdonald and Engelhardt developed deterministic functions for each of these processes based on physical laws (Macdonald and Engelhardt, 2003). There are many other natural phenomena that evolve with the same birth, growth, and death phases. Virtually any biological entity (i.e., tree growth in a forest) as well as many other objects (i.e., fire spread in a region) can develop through the same three phases. Most models of corrosion focus on a single phase of evolution, and of those, initiation and growth are most frequently observed. Any complete model of corrosion evolution needs to account for all three phases as well as the spatial characteristics of the process in order to provide a meaningful prediction over time.

2.2.1. Point Processes

Corrosion events are initiated at various locations on a metallic surface. For the purposes of this research, corrosion is observed using images from non-destructive inspection or other methods that produce digital images of the surface. Therefore, corrosion initiation occurs at a given pixel location and point in time. A point process generates a set of event locations, or a point pattern, which are irregularly distributed within a given space (e.g. geographic or temporal) (Diggle, 1983).

2.2.1.1. Spatial Point Process

A spatial point process determines the distribution of events in a geographic space. For a given data location, $s \in \Re^d$, the general definition of a spatial process is given by $\{Z(s) : s \in D\}$, where *D* is random, a spatial point process. *D* can be thought of as a collection of random events, and a realization of *D* is a spatial point pattern (Cressie, 1993).

Typically, analysis of a spatial point process begins with a determination of complete spatial randomness (CSR). CSR implies a homogeneous Poisson process such that events are uniformly distributed over a bounded region, *A* (Cressie, 1993). Under CSR, the intensity of the events, λ, does not vary over the space, and there are no interactions among events (Diggle, 1983). There are many examples in the literature of spatial point patterns using quadrat, distance, and K-function methods to analyze a wide range of data (Fisher, 1922; Clark and Evans, 1954; Ripley, 1977).

Quadrat analysis involves counting the number of events that occur within each quadrat and comparing the result with what would be expected if the process were indeed CSR (Cressie, 1993). Many variations for this method have been developed, mostly to overcome the inherent shortcomings of determining the size and location of quadrats. Quadrats are typically placed randomly over the region, but the size of the quadrat can significantly impact the assessment of CSR. Quadrat methods can provide an acceptable test for CSR in a field environment but cannot provide a spatially informative analysis of a point pattern because precise spatial information on the locations of events is ignored (Cressie, 1993).

Distance methods provide another means for determining CSR in a spatial point pattern. Because they utilize event locations, distance methods require a precise mapping of the events (Cressie, 1993). Examples of distance methods include nearest-neighbor techniques where event-to-event distances, as well as event-to-point distances are considered. The distribution theory for various nearest-neighbor distances is well established and can therefore be easily used for determination of CSR (Cressie, 1993). Quadrat and distance methods can determine if a pattern exhibits CSR but are unable to quantify the amount of departure from CSR.

The K-function can be estimated in order to capture the spatial dependence between regions of a point process. In simple terms, the K-function measures the number of events within a distance, *h*, of an arbitrary event. While the intensity function, λ, measures the first-order properties, the K-function characterizes the second-order properties of a spatial process [36]. Given the set of events $\{s_{1,} \ldots, s_N\}$ in a region *A*, *N(A)* is the number of events in *A*, $\|s_i\text{-}s_j\|$ is the norm function which represents the distance from event location s_i to s_j, and $I(x \leq h)$ is the indicator function which = 1 if $x \leq h$, the K-function can be defined as follows (Cressie, 1993):

$$\hat{K}(h) \equiv \hat{\lambda}^{-1} \sum_{i=1}^{N} \sum_{j=1}^{N} I\left(\left\|s_i - s_j\right\| \leq h\right) / N(A) \qquad (1)$$

2.2.1.2. Spatial-Temporal Point Processes

Many spatial processes are also dynamic over time. A great deal of research is available on models of spatial-temporal processes such as rainfall cells (Smith and Karr, 1985), earthquakes (Fiksel, 1984), and criminal incidents (Liu and Brown, 2003). There are two classes of spatial-temporal point processes, which are based on the duration of the events over time (Cressie, 1993). A space-time shock point process generates events instantaneously over time and space. A space-time survival point process generates events at a random time and location, which survive for a random length of time. Earthquakes, crimes, and corrosion events are examples of space-time shock point processes. A survival process would be applicable to most ecological or biological events.

A major portion of the literature in spatial-temporal prediction is in the development of the space-time autoregressive moving average (STARMA) models (Cliff *et al.*, 1975; Pfeifer and Deutsch, 1980). STARMA is a generalization of the autoregressive moving average (ARMA) model for time series data combined with spatial models. The STARMA class of models is characterized by linear dependence lagged in both time and space (Pfeifer and Deutsch, 1980).

2.2.1.3. Spatial-Temporal-Feature Model

Spatial-temporal point patterns are often distinguished as a marked space-time process (Cressie, 1993). In addition to a time and location, each event is characterized by a randomly distributed mark or measurement associated with the event. For corrosion events that occur at a given location and time, example marks could be the distance to the nearest rivet, environmental conditions, or the number of corroded neighboring cells. These marks represent event-related feature information which can be exploited for spatial-temporal prediction. Feature-based modeling has proven to be effective in the prediction of other spatial point processes, such as the prediction of crime incident locations (Liu, 1999; Liu and Brown, 2003).

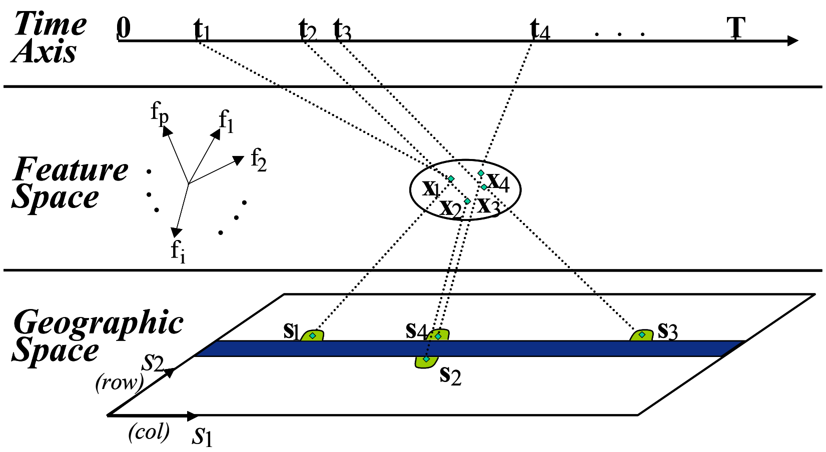


Figure 4. Event occurrence in time, feature, and geographic space

The marks, or features, of a spatial-temporal point process provide information on the tendency of the process to produce an event at a given time and location. Events occur in time, geographic and feature space. If the features accurately describe event initiation, then clusters in feature space should be evident regardless of their proximity in time or geographic space (Liu, 1999). Figure 4 is an illustration given by Liu to illustrate the expected clustering in feature space of spatial-temporal events. Geographical and temporal locations that share feature space with event occurrences are "at risk" for a future event occurrence. The rationale for incorporating feature space into space-time prediction models is to identify potential areas for event occurrence (Liu, 1999). In Liu's application, features were used to identify locations for incidents of crime, and provided an improvement upon purely geographic methods.

2.2.2. Classification Models

The data used in this research is derived from time sequenced digital images of corrosion growth in a sample material. At each time, the pixels in each image are classified as being corroded or non-corroded. The feature values at each pixel location are used to predict the occurrence of corrosion. The overall goal is to determine the best set of features and the appropriate functional form to accurately describe the corrosion growth process. There are many supervised learning methods available for modeling the occurrence of corrosion as a function of the feature values. An important consideration in model selection, however, is the large data set for this application. Brief descriptions of various methods as well as examples of their use in the literature are given in the following sections. More detail and description for use in this application is provided in Chapter 3 for selected methods.

2.2.2.1. Logistic Regression

Logistic regression is a linear method for classification similar to linear discriminant analysis and ordinary least squares (OLS) regression. Logistic regression is a popular method for classification because it does not require some of the more restrictive assumptions of OLS (Hosmer and Lemeshow, 2000). Specifically, logistic regression does not require the dependent variables to be normally distributed, which is an assumption typically not applicable to most applications. Additionally, logistic regression does not assume homoscedasticity in the variance of the response variable across values of the predictor variables.

Logistic regression models have been applied to a wide range of application fields including ecology, maintenance inspection, and medicine (Ariaratnam *et al.*, 2001; Allen and Lu, 2003; Zhou *et al.*, 2005). Variations on the standard logistic regression model have been published in order to address the assumptions inherent in the model, such as the assumption of independent observations (Bonnie, 1987). Logistic regression remains a popular choice for classification problems and is primarily chosen for applications where the goal is to determine the relationship between the predictor variables and the response variable (Hastie *et al.*, 2001).

2.2.2.2. Classification and Regression Trees

Tree-based models provide a simple and powerful method for regression and classification. The method of constructing classification and regression trees (CART) through the use of recursive partitioning was introduced by Breiman, *et al.* and provides methods for creating decision trees for categorical or continuous response variables (Breiman *et al.*, 1984). Like logistic regression, CART is widely used in many application fields because of the ability to interpret the relationship between predictor and response variables, although complex trees can hinder this ability. However, unlike logistic regression, CART handles different variable types and requires no prior knowledge of nonlinear or complex relationships between predictor variables (Hastie *et al.*, 2001). The primary limitation of trees is the high variance that can result from small changes in input data that causes differences in tree splits. Bagging, or ensemble trees, can alleviate this problem by averaging the results across multiple trees (Breiman, 1996a)

CART methods have been applied to corrosion prediction applications (Brence and Brown, 2002a). The goal of the model was to remove subjectivity in classifying eddy current scans of corrosion in aircraft using CART as a means of classifying corrosion damage based on the voltage measurements. Brence and Brown found that regression tree models outperformed many other standard model types, including logistic regression.

2.2.2.3. Neural Network Classifiers

Classification can also be accomplished using a neural network model. A neural network acts as a black box that maps the set of predictor variables to the set of expected response variables. Neural networks work as a two-stage classification scheme where linear combinations of the predictor variables are generated and the response is then modeled as a nonlinear function of the predictor combinations. Neural networks are primarily suited for applications with a high signal-to-noise ratio and can be very effective for predictive results; however, the black box nature of the methodology does not provide any insight into the relationship between the predictor and response variables (Hastie *et al.*, 2001).

Because of the ability to produce accurate predictions of complex systems and their availability in commercial software packages, neural networks have grown in popularity among practitioners in various fields. There are numerous examples of the use of neural networks in corrosion prediction applications (Cai *et al.*, 1999; Leifer *et al.*, 1999; Pintos *et al.*, 2000; Nesic *et al.*, 2001). Most of these efforts endorse the use of neural networks for their ease of use and ability to capture the highly non-linear relationships between predictor and response variables. The primary goal of these models is to predict corrosion rates in various materials; however, the inability of neural networks to give insight into this relationship makes it a less attractive alternative for this research.

2.2.2.4. Kernel Density Classifiers

Another unsupervised learning procedure which can be used for classification is kernel density estimation (KDE). KDE seeks to estimate the probability density, $f_X(x)$, for some random sample $x_1,\ldots, x_N$ by averaging a known density function across the data observations. The averaging of the kernels produces a smooth approximation to the true density.

The major drawback to KDE is it's reliance on the entire training data set. The KDE model is the data set, and fitting of the model is done at evaluation (Hastie *et al.*, 2001). Because of computational visualization considerations with KDE, we chose to reduce the dimension of the data set. Principal component analysis (PCA) provides an accepted method for shrinking the number of features and can simplify KDE (Scott, 1992; Cooley and MacEachern, 1998). In addition, it ensures that regardless of any correlation in the original feature variables, the new principal component features will be uncorrelated.

2.2.2.5. Other Methods

Many other supervised learning methods can be used for classification problems. Methods such as generalized additive models (GAM) are built upon the same basic principle as linear regression. Linear regression, however, cannot easily capture the nonlinear relationships that typically exist in most applications. GAMs utilize a generalized smooth function in place of the linear terms. GAMs are typically more flexible than linear methods while still providing insight into the process; however, this flexibility makes it difficult for most of the fitting methods to handle large data sets (Hastie *et al.*, 2001).

Another applicable model is multivariate adaptive regression splines (MARS). MARS is a generalization of the CART method modified to improve regression performance. MARS uses piecewise linear basis functions with knots at each observed value for each input. Initially, this results in over fitting of the data, so a backwards deletion procedure is used to sequentially remove the basis function that causes the smallest increase in residual error. MARS is especially useful for localized approximation functions in high dimensions (Hastie *et al.*, 2001).

Support vector machines (SVM) are an extension of the linear decision boundary for classification where a hyperplane is used to separate the data into multiple classes. In most cases, classes are not linearly separable due to overlap. SVM maps the input data into a high dimensional feature space where the data is more separable. SVM uses only the points closest to the boundary and seeks to achieve the widest separation between classes and from each class to the boundary. Like neural networks, SVM is a popular tool because of its ability to handle highly nonlinear relationships and high dimension data. SVM has been used a great deal in extremely complex applications such as gene classification (Brown *et al.*, 1999). However, the transformation of predictor variables into a feature space makes interpretation of the results and insight into the process difficult (Hastie *et al.*, 2001).

2.2.3. Growth Models

Most biological forms can be represented as a stochastic growth process (Edelman, 1988). The natural laws that govern the growth of the object are also subject to a random disturbance. The random element is a source of inexactness that causes two realizations of the same process to be different. Despite this difference, parameters that are estimated from these realizations should be consistent as the size of the realizations increases (Cressie and Laslett, 1987). Much of the research on growth models involves determining an asymptotic shape for the modeled object. The shape, along with the natural laws that govern the process, assists in the selection of an appropriate model form.

Many growth models focus on the evolution of an object on a grid. Growth occurs as grid locations, or cells, become "activated". The aggregation of the activated cells represents the object at a given point in time. This methodology is explained as a random closed set growth model. Growth models are commonly classified by their initial configuration, cluster properties, and particle formation (Cressie and Laslett, 1987). The initial configuration is used to describe processes that begin from a single cell or from multiple cells. Cluster properties describe the growth rules for the object, which can be determined by local or global cluster properties. Additionally, some processes produce compact clusters while others form clusters of fractal dimension. Particle formation refers to the manner in which new cell growth occurs. New cells are either produced from within, or initiated externally and then attached to the cluster.

2.2.3.1. Random Sets

In general, a random set can be thought of as a special case of a random function that takes on values of 0 or 1 (Cressie, 1993). The boundary of the random set is formed by looking down on the random function and slicing it at a given level. The theory behind random sets (Matheron, 1975) is based on a hit-or-miss concept where the set is completely characterized by repeatedly intersecting it with an arbitrary compact test set and tracking whether the intersection is empty or not. Let *E* be a locally compact, countably dense, Hausdorff space. A space *X* is considered a Hausdorff space if for a distinct $x,y \in X$, there are disjoint open sets U,V $(U \cap V = \varnothing)$, such that $x \in U$, $y \in V$. Suppose $\mathcal{F}$ is the set of all closed subsets of *E* (e.g. $E = P^D$). The subsets of *E* which "hit" F are represented as $\mathcal{F}_A \equiv \{F \in \mathcal{F}: F \cap A \neq \varnothing\}$, while the subsets which "miss" F are represented as $\mathcal{F}^A \equiv \{F \in \mathcal{F}: F \cap A = \varnothing\}$ (Molchanov, 1997; Molchanov and Scherbakov, 2003).

A random closed set model (RACS) places sets (i.e. a collection of points such as a disk or a line) at random positions to form a random subset of a plane or space. The plane or space is then divided into points which are members of the random set and those that are not. Collectively, the points in the random set represent the object of interest. The Boolean model is a simple example of a RACS model. Independent and random sets, Z, are produced from a distribution of sets. These sets are placed at locations, s_i, within a domain D determined by a homogeneous Poisson point process. The points of the Poisson process are called foci or germs. The sets at each germ are called grains. The Boolean model, *X,* is therefore defined as the union of the grains at each germ.

$$X \equiv \cup\{Z(s_i) : s_i \in D\} \tag{2}$$

RACS growth models have been used to predict the evolution of objects such as *in vitro* tumors (Cressie and Hulting, 1992). The tumor at time 1 is grown by placing closed sets Z at locations, s_i, which are points within the tumor at time 1 chosen by a Poisson process. Specifically, X_2 represents the tumor at time t_2, such that $X_2 = \cup\{Z(s_i) : s_i \in G\}$ where G is a non-homogeneous Poisson process with intensity function $\lambda(s_i)$.

$$\lambda(s) = \begin{cases} \lambda, s \in X_1 \\ 0, \text{otherwise} \end{cases} \tag{3}$$

Figure 5 shows the progression of the simulated tumor from a single disk to an irregular set using fixed radius grains, *Z* (Cressie and Hulting, 1992). Grain disks of fixed or random radius are placed according to the relationships defined above. Each set is therefore generated from the previous set. Because of the dramatic change in growth rates over time, the model does not provide accurate representations of tumor growth over extended periods. For short time horizons the fitted model agrees with experimental data.

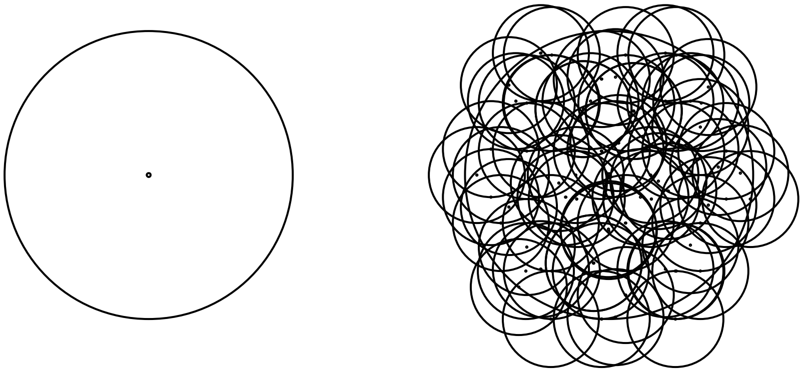


Figure 5. Example realization of tumor growth model

Random set models provide a method for modeling growth when more than a summary statistic is needed; however, implementation of the model can quickly become intractable (Serra, 1982). Specifically, the model is completely defined by the hitting function, T_X. Unfortunately, estimation of this function and its parameters is only manageable for simplistic models (Cressie and Laslett, 1987). For a random closed set *X*, and a compact set *K*:

$$T_X(K) = \Pr\{X \cap K \neq \varnothing\} \qquad (4)$$

Many extensions of the Boolean model have been developed. The primary difference in the models is the specification of the point process which controls the locations of the foci and grains (Neyman and Scott, 1958; Serra, 1980; Mase, 1982). In particular Neyman and Scott utilized a parent-child point process which generated parent points according to a homogeneous Poisson process and generated independent and identical child points around each parent location (Neyman and Scott, 1958). Other prevalent growth models, such as contact processes, Eden models, and Conway's game of life are all special cases of the random set growth model.

2.2.3.2. Contact Processes

Interacting particle systems are spatial configurations of particles whose states change probabilistically according to the states of neighboring particles. Variations of interacting particle models have been used to model a wide range of complex processes (Rikvold, 1982; Hermann, 1986; Greene, 1989; Griffeath, 1993; Hermann, 1996; Liggett, 1997; Ivanenko *et al.*, 1999; Halsey, 2000; Hug *et al.*, 2002). The contact process is the simplest form of interacting particle systems (Thompson, 1994). Contact processes are local growth models where the activation of a cell depends solely on the status of its neighbors. Generally, contact processes are continuous time models. The initial cell is activated at time t=0, and subsequent cells become activated at exponential rate, λ, which is proportional to the number of neighboring active cells. Therefore, in a four-neighbor structure the activation rate is either 0, λ, 2λ, 3λ, or 4λ. Many discretized variations of the continuous-time contact process have been developed using small time intervals (Cressie, 1993).

Epidemic models are discretized contact processes and have been used to model phenomena such as the spread of disease or fire (Thompson and Rosenfeld, 1999). The most basic epidemic model is also referred to as site percolation and is developed using the following algorithm:

1. Activate originating cell at time t=0
2. At each t_i, inactive neighboring cells are activated (simultaneously) with probability p, and blocked with probability (1-p) (blocked cells are never activated)

There is a critical probability value for p, such that lower probabilities result in a finite-sized cluster, whereas higher values of p almost surely result in infinite-sized clusters (Thompson, 1994).

Growth models can also be based on random walks where the walk begins at the origin and visited points are included in the cluster of active cells. The walk moves randomly to one of the four-neighbor cells. Some applications of the random walk growth model allow for the same cell to be visited twice, whereas others restrict the walk to be self-avoiding. Growth processes modeled using random walks are called kinetic growth models and typically result in a compact and non-isotropic cluster(Thompson and Rosenfeld, 1995; Aristoff, 2003).

2.2.3.3. Eden Growth Process

In global growth models the probability of activating a cell at a given point is dependent on the structure of the entire cluster. The Eden growth model (Eden, 1958) is an example of a global growth model where the probability of activation for a cell is determined by the number of cells in the boundary of the cluster. The following is a simple algorithm for the Eden growth process:

1. At t=0, activate a cell at the origin
2. At each t_i, randomly activate a single cell on the 4-neighbor cluster boundary

The expected shape of clusters produced by the Eden model is difficult to predict. Some studies have suggested that the limiting shape is circular, and other results show a flattening in the diagonal directions. Figure 6 provides an example of an Eden cluster of 500 cells (Thompson, 1994). The random growth of the Eden model has been shown to closely relate to the growth of bacterial colonies, tissue cultures, and tumors (Thompson, 1994).

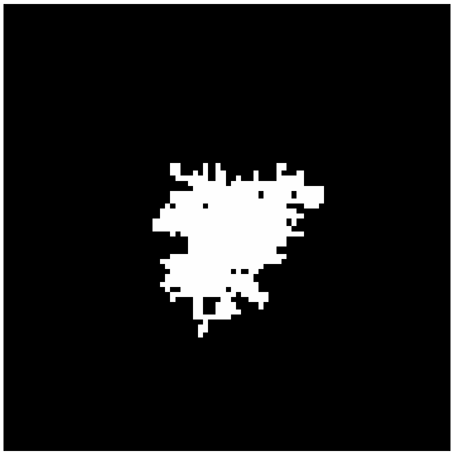


Figure 6. Eden cluster model after 500 iterations

CHAPTER 3. METHODOLOGY

The third chapter describes the methodology used to predict the location and time of corrosion initiation and growth. The formal definitions and notation are specified for both the initiation and growth models. Next, the use of classification models to determine the growth probability function is explained. Finally, a summary of the data processing of time-sequenced images used in the application of this methodology is given.

3.1. Problem Definition

There is a significant need to accurately model and predict the evolution of corrosion. Given image data of corrosion events within an area of a material and at discrete points in time, we seek to determine if the features inherent in these images can be employed in a feature-based model to predict the initiation and growth of filiform corrosion. Additionally, we evaluate the effectiveness of the feature-based approach in spatial and temporal growth prediction.

3.2. Model Development

The initiation of a corrosion event and the corresponding growth around the initiation location is similar to the germ-grain model of random closed sets. Corrosion initiation occurs at a given location, and growth occurs from the point of initiation. As a germ-grain model, the initiation of corrosion is the germ, and the growth at that location is the grain. For a given piece of material, each geographic location has inherent features which influence the likelihood of corrosion initiation or growth at each site. The process which controls the location of initiation and the direction and magnitude of growth can best be captured in a feature-based model that accounts for these differences. Traditional methods overlook these features and only explain the occurrence of events or growth of an object with respect to space and time. Features inherent to corrosion evolution include environmental conditions such as humidity and temperature, as well as more location-specific features such as distance to rivets (or other inclusions/imperfections), distance to corroded sites, and number of neighboring locations with corrosion. A corrosion evolution model can be decomposed into a feature-based point process model for initiation and a feature-based growth model.

3.2.1. Initiation

The initiation of corrosion on a finite region of material can be modeled as a point process for event occurrence. Liu and Brown introduced a feature-based point process model for the prediction of criminal incident locations in space and time (Liu and Brown, 2003). This model proved very effective in predicting potential event locations and outperformed traditional geographic-based approaches for most test cases. The premise of the model is that the selected features capture the data structure produced by the underlying event generation process. The primary difference in the application to criminal incidence versus corrosion initiation is that crimes can occur at the same location at different times. Once a site is corroded, however, it remains corroded.

Denote the locations and times of corrosion events as $\{(s_1,t_1), (s_2,t_2), \ldots, (s_n,t_n)\}$, where $t_0=0<t_1<t_2<\ldots<t_n$. The corresponding features for each location-time pair is x_{s_1,t_1}, x_{s_2,t_2}, …, x_{s_n,t_n}, which represent a realization of a marked space-time shock point process because events are generated instantaneously over time and space and are marked by the features that characterize them. Corrosion growth may be seen as a survival process due to the fact that growth only occurs as long as the electrochemical environment can support growth. Initiation, however, is a shock process because it is instantaneous and remains at the location. The form of the process is defined below, where t, s, and $x_{s,t}$ are random. Events occur at coordinate locations, $s=(s_x,s_y)$, within a region $D\subset\Re^2$. $T\subset\Re^+$ is the collection of times when events occur (study horizon), and $\chi\subset\Re^p$ is the collection of values for the p-dimensional feature vectors (feature space).

$$\{x_{s,t} \in \chi : s \in D, t \in T\} \tag{5}$$

The development of the following initiation model is derived from Liu's model for criminal incidence prediction (Liu and Brown, 2003). Suppose we observe $T_n=\{t_1, t_2, ..., t_n\}$, $D_n=\{s_1, s_2, ..., s_n\}$, and $\chi_n=\{x_1, x_2, ..., x_n\}$, where $x_i = x_{s_i,t_i}$, we seek to estimate the transition density for the location and time for the next event occurrence $s_{n+1} \in D$, and $t_{n+1}>\mathrm{t_n}$, where $\nu(ds_{n+1})$ is the area of the infinitesimal region around s_{n+1} and dt_{n+1} is the infinitesimal time horizon t_{n+1}. The transition density represents the likelihood that the next event occurs within the infinitesimal location region and time horizon, given the locations, times, and features of all previous events.

$$\psi_n(s_{n+1},t_{n+1} \mid D_n,T_n,\chi_n) \equiv \lim_{\nu(ds_{n+1}),dt_{n+1}\to 0} \frac{\Pr\{N(ds_{n+1},dt_{n+1})=1 \mid D_n,T_n,\chi_n\}}{\nu(ds_{n+1})dt_{n+1}} \qquad (6)$$

If we assume that the feature space, χ, does not contain temporal features and that the temporal transition of the point process is independent of the spatial transition, then the transition density, $\psi_n(s_{n+1},t_{n+1} \mid D_n,T_n,\chi_n)$, can be decomposed into the spatial transition density, $\psi_n^{(1)}$, and the temporal transition density, $\psi_n^{(2)}$. These assumptions are reasonable if we agree that the intensity of corrosion occurrence at a location does not affect the time at which the next event occurs.

$$\psi_n(s_{n+1},t_{n+1} \mid D_n,T_n,\chi_n) = \psi_n^{(1)}(s_{n+1} \mid D_n,T_n,t_{n+1,}\chi_n) \cdot \psi_n^{(2)}(t_{n+1} \mid T_n) \qquad (7)$$

The development of the spatial transition density is dependent on the assumption that the features accurately capture the behavior of the true event generation process. Although the corrosion initiation events will not necessarily be clustered in geographic space, clustering is expected in feature space. Clustering in feature space reveals the combinations of feature values that have a higher propensity to initiate a corrosion event. If no clustering occurs, then the features do not influence the choice of location. However, the literature supports the assumptions that corrosion initiation is dependent on such features as temperature, humidity, other corroded sites, and proximity to inclusions.

The feature space analysis can be used to model the first-order effects (event intensity) of the spatial point process. The feature values at a geographic location determine the potential for corrosion initiation to occur at the site. These preferential combinations should continue to hold from t_n to t_{n+1}, which describes a stationary process. The second-order effects are modeled in geographic space to describe the interaction between locations. Therefore, the spatial transition density, $\psi_n^{(1)}(s_{n+1} \mid D_n, T_n, t_{n+1,}\chi_n)$, can be defined where α is a normalizing constant, $\psi_n^{(11)}(x_{n+1} \mid \chi_n)$ is the first-order spatial transition density, and $\psi_n^{(12)}(s_{n+1} \mid D_n, T_n, t_{n+1})$ is the second-order spatial transition density. Estimation of the model components and designation of significant features is accomplished using data for corrosion evolution.

$$\psi_n^{(1)}(s_{n+1} \mid D_n, T_n, t_{n+1,}\chi_n) = \alpha\psi_n^{(11)}(x_{n+1} \mid \chi_n)\psi_n^{(12)}(s_{n+1} \mid D_n, T_n, t_{n+1}) \qquad (8)$$

3.2.2. Growth

The model for corrosion initiation can be used to produce a probability field for the location of the next initiation event given the feature values inherent to each location. This field can then be used to determine the location of the next simulated corrosion initiation event. Once the event location is determined, an interacting particle model can be used to simulate corrosion growth in space and time.

The growth of filiform corrosion is easily adaptable to an interacting particle system. The filament initiates at a given location and then subsequent locations become corroded according to the growth process. In an interacting particle system, this growth process is assumed known and is represented as a transition function which determines the state of the system. The growth model for this research is an interacting particle system represented by the dynamic, discrete time, discrete state system: $\{L, \mathcal{C}, \Gamma, g\}$ (Liggett, 1985).

L: finite graph with vertex set $\mathcal{V}$, such that $|\mathcal{V}| = \mathrm{I}$. Typically, L is a d-dimensional regular lattice where each vertex $\mathrm{i} \in \mathcal{V}$ is assigned a state value $\mathrm{c}_\mathrm{i} \in \mathcal{C}$

$\mathcal{C}$: finite set of possible vertex states. Typically, $\mathcal{C} = \{0,1\}$

Γ: $\mathcal{V} \rightarrow \mathcal{V}$, neighborhood function which defines the finite set of neighborhood vertices. For a graph L, the neighborhood of vertex i, denoted $\Gamma(\mathrm{i})$, is the set of vertices adjacent to i. If the vertex i is included, then $\Gamma(\mathrm{i}) \cup \{\mathrm{i}\}$ is the closed neighborhood of i. The neighborhood of a given vertex $\mathrm{i} \in \mathcal{V}$, is denoted $\gamma = \Gamma(\mathrm{i})$.

g: $\mathcal{C}^\gamma \rightarrow \mathcal{C}$, the local transition function defined by a transition table or rule set. The function g calculates the future value of the state of a single vertex from the current states of the neighborhood vertices. For discrete time, the future state of vertex i at time $\mathrm{t+1}$ is a function of the states of the neighborhood vertices at time t: $\mathrm{c}_\mathrm{i}^{\mathrm{t+1}} = \mathrm{g}\left(\mathrm{c}_{\Gamma(\mathrm{i})}^\mathrm{t}\right)$.

The global state of the system at time t, $\mathrm{C}^\mathrm{t} \in \mathcal{C}^\mathcal{V}$, is a string configuration of the state of the system at each vertex i at time t. The global transition function $\mathrm{G} : \mathcal{C}^\mathcal{V} \rightarrow \mathcal{C}^\mathcal{V}$, translates the current configuration C^t to the next configuration $\mathrm{C}^{\mathrm{t+1}}$. G is defined by evaluating the local transition function g at each vertex i.

$$\mathrm{C}^\mathrm{t} = [\mathrm{c}_1^\mathrm{t}, \mathrm{c}_2^\mathrm{t}, \ldots, \mathrm{c}_\mathrm{I}^\mathrm{t}] \tag{9}$$

$$\mathrm{G} = [\mathrm{g}\left(\mathrm{c}_{\Gamma(1)}\right), \mathrm{g}\left(\mathrm{c}_{\Gamma(2)}\right), \ldots, \mathrm{g}\left(\mathrm{c}_{\Gamma(\mathrm{I})}\right)] \tag{10}$$

This system definition is a general version of a typical interacting particle model where state transitions at a given vertex are dependent on the states of the vertices in the defined neighborhood. In many processes, however, the evolution of the system is influenced by features other than the states of neighboring vertices. If we consider the growth of a plant as depicted in time sequenced images, we can model this as an interacting particle system, where $\mathcal{C} = \{0,1\}$, 0: the plant does not occupy a given pixel, 1: the pixel is part of the plant (Greene, 1989). The probability that a given vertex (or image pixel) transitions from 0 to 1 is dependent on the states of the neighboring vertices because growth occurs only from the areas currently occupied by the plant. However, other factors, such as the amount of sunlight available at that pixel location, can also influence the likelihood of transition from $c_i = 0$ to $c_i = 1$. In order to account for these features, the previous system definition becomes $\{L, \mathcal{C}, \Gamma, p, f\}$

p : $\chi \to \Re^+$, where $p(x_i)$ is the probability that growth occurs as determined by the feature values at vertex i.

f : $\mathcal{C}^\gamma \times \Re^+ \to \mathcal{C}$, the local transition function utilizing the states of neighboring vertices and the probability of growth determined by feature values at vertex i. The future state of vertex i at time $t+1$ becomes $c_i^{t+1} = f\left(c_{\Gamma(i)}^t, p(x_i)\right)$.

The model components above can be estimated using a variety of methods. Of particular interest is the specification of the transition function, f. In standard interacting particle models this function is described by a rule set that is defined by knowledge of the system process. In the original Ising model for ferromagnetism, vertices transition from a state of magnetization, based on the magnetization of their neighbors. The rules governing this transition were understood and could be included in the model along with a tuning parameter representing system temperature which influences transition. Changes in temperature produce different configurations in the system.

Although very little is known about the features that influence the growth of filiform corrosion, certain observations can be made regarding the mechanics of filiform movement. For example, filaments initiate at points along defects in the coating. The filament grows in a worm-like fashion with a visible head acting as the frontier of growth. As the head traverses under the coating the filament body behind it corrodes outward causing the filament to spread. The spread continues until the corrosion reaction can no longer be maintained. Another characteristic of filament growth is the self-avoidance of the filament path. Filaments progress along uncorroded areas, so they do not cross over their previous path. These characteristics can be incorporated into a simplified example using the previously defined model, in order to show the applicability of the model to capture the characteristics of filiform growth.

Assume a regular lattice defined by a 201 x 201 pixel image, with vertex state set $\mathcal{C} = \{0,1,2,3\}$, where 0: uncorroded, 1: head cell-directing growth, 2: active cell-continuing growth, and 3: corroded. Assume a single filament initiates at the center pixel location (101,101). In order to represent the growth probability function, p, a kernel density estimate is created over the area with given "events" located in the northeast corner of the surface. This produces a probability field over the area that causes pixels closer to the northeast corner to have a higher probability of corrosion than pixels in other locations. The representation of the resulting p function is provided in Figure 7, with higher probability represented as higher intensity (lighter) pixels.

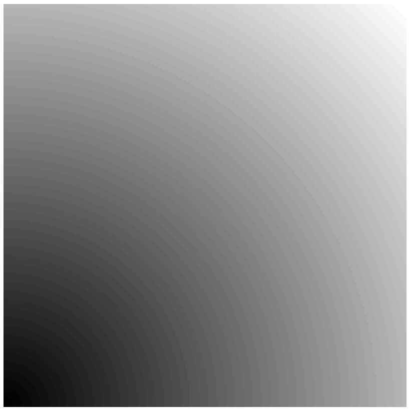


Figure 7. Representation of example growth probability function

Using the previous characteristics and the growth probability function, the growth of the filament can be simulated over the given region. Variables are included in the simulation to control the growth of the filament. The width of the filament is controlled by a variable, *Spread*, which determines the number of iterations that the filament grows after the head moves through an area. A higher value of *Spread* causes the corrosion process to continue longer and results in a wider filament. Another variable, *Dprob*, is used to control the dependence of the movement on the underlying growth probability function, p. A *Dprob* value of 1 forces the filament to proceed in the direction of higher probability, which in this case would result in growth following a straight line to the northeast corner. A *Dprob* value of 0, allows the direction of growth to be completely determined by the underlying probability values, in other words in a random fashion where each pixel is weighted by its probability value.

Realizations from this example model are shown in Figure 8. A *Spread* value of 4 pixels was used for both realizations, which causes cells within a range of 4 to corrode once the head cell moves through a given area. The first realization was generated using a *Dprob* value of 0.2. This low value causes the filament to meander around the area, while maintaining general movement in the northeast direction. The second realization uses a *Dprob* value of 0.8 which forces the filament to track due northeast 80% of the time. The purpose of this example is to show the ability of the previously described growth model to adequately capture the growth characteristics of a single filament of corrosion. The direction of growth is primarily controlled by the underlying growth probability function, which for the purposes of this example, is assumed to be correct.

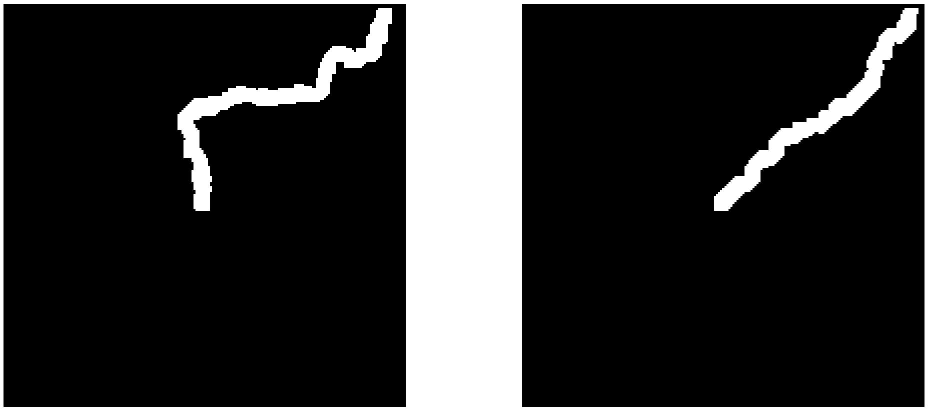


Figure 8. Realizations from example growth model (left: Dprob=0.2, right: Dprob=0.8)

3.3. Growth Probability Function Models

The growth probability function, p, determines the probability that a given pixel corrodes based on the values of the features at the pixel location. This probability value is used by the local transition function, f, to evolve the corrosion states of pixels. In the previous example, uncorroded pixels that neighbor the current head cell can transition to a head cell state. The probability of this transition is determined by the growth probability value. For *Dprob*=1, the neighboring pixel with the highest growth probability value becomes the new head cell, with ties resolved by a random selection. Any of the classification models discussed in Chapter 2 can be used to determine the probability of growth based on feature data. Due to the size of the datasets used in this research and preliminary results for various model types (discussed further in Chapters 4 and 5), logistic regression and CART models are the primary alternatives considered. Regardless of the method selected, we seek to determine the relationship between the predictor and response variable. The preferred model is the parsimonious model that provides the best fit of the data.

3.3.1. Logistic Regression

Logistic regression is similar to a standard linear regression model, except that the response variable is dichotomous. For the purposes of this research, we are concerned with a vector of binary responses, $\mathbf{y}$, where $y_i = 0$ implies that pixel i is not corroded, and $y_i = 1$ implies pixel i is corroded. We can consider the observation of each vector of feature values at a pixel location, $\mathbf{x_i}$, as a Bernoulli trial. Therefore, y_i is a Bernoulli random variable with mean parameter $\mu(\mathbf{x_i}) = \Pr(y_i = 1|\mathbf{x_i})$. We seek to find a linear model for some function, or link, for μ. One approach is to use a logistic function, where $\boldsymbol{\beta}^T$ is the transpose of the vector of predictor variable coefficients.

$$\mu(\mathbf{x_i}) = \frac{e^{(\boldsymbol{\beta}^T\mathbf{x}_i)}\ ^{t}}{1+e^{(\boldsymbol{\beta}^T\mathbf{x}_i)}\ _{I}} \tag{11}$$

The resulting model equation becomes: $y_i = \mu(\mathbf{x_i}) + \varepsilon_i$, where ε_i is the error at observation i. The shape of the logistic function can be seen in Figure 9 for a single feature. Due to the binary response variable, the value of ε_i will be $1-\mu(\mathbf{x_i})$ when $y_i = 1$, and $\varepsilon_i = \mu(\mathbf{x_i})$ when $y_i = 0$. Given the Bernoulli distribution of y_i, $E[\varepsilon_i] = 0$ and $\text{Var}(\varepsilon_i) = \mu(\mathbf{x_i})(1-\mu(\mathbf{x_i}))$. The distribution of the error violates the linear regression assumption of homoscedastic error terms.

The resulting logistic regression model is a non linear function of $\boldsymbol{\beta}$, therefore we can transform the logistic model using the logit link function $f(\mu) = \log(\mu/(1-\mu))$. The logit function takes the logarithm of the odds that $y_i = 1$. Applying the logit function to the response variable yields a model linear in the coefficients, however, because of the distribution of the error terms, we must estimate the coefficients using a method such as maximum likelihood or iteratively re-weighted least squares. From this model, we can determine the predicted probability that a pixel corrodes as a function of its feature values.

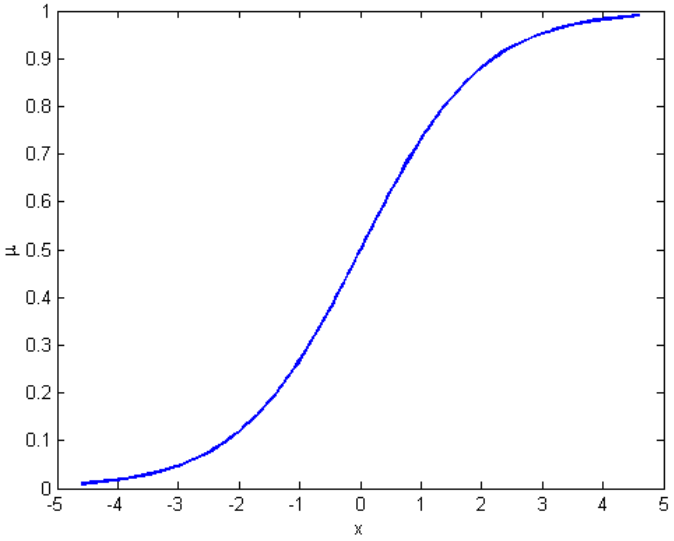


Figure 9. Example logistic function for a single feature

3.3.2. Classification and Regression Trees

Classification trees are a simple, but powerful classification technique that can work well across a wide range of applications. The process of constructing a classification tree for a given set of data consists of two primary tasks. The first is to recursively partition the predictor space, and the second is to prune the resulting tree.

For a given data set of $i = \{1,2,\ldots,n\}$ observations, we have values of p predictor variables, X_j, where $j = \{1,2,\ldots,p\}$. Recursive partitioning seeks to divide up the p-dimensional space of X into non-overlapping hyper-rectangles. Figure 10 shows an example of recursive portioning for two predictor variables with a binary response. The data is first partitioned at point s1 for variable X_1, then the splits are determined for variable X_2 at s2 and s3. The goal is to ensure that each hyper-rectangle region is as pure as possible, meaning that the region contains observations belonging to just one class. Of course, this means that we could continue to divide up the space until each observation is contained in its own region. Although this would ensure perfect classification, the model would over fit the data and be of little use in predicting a response with a new data set.

Therefore, a minimum node size is used to stop dividing a region once it contains a specified number of observations.

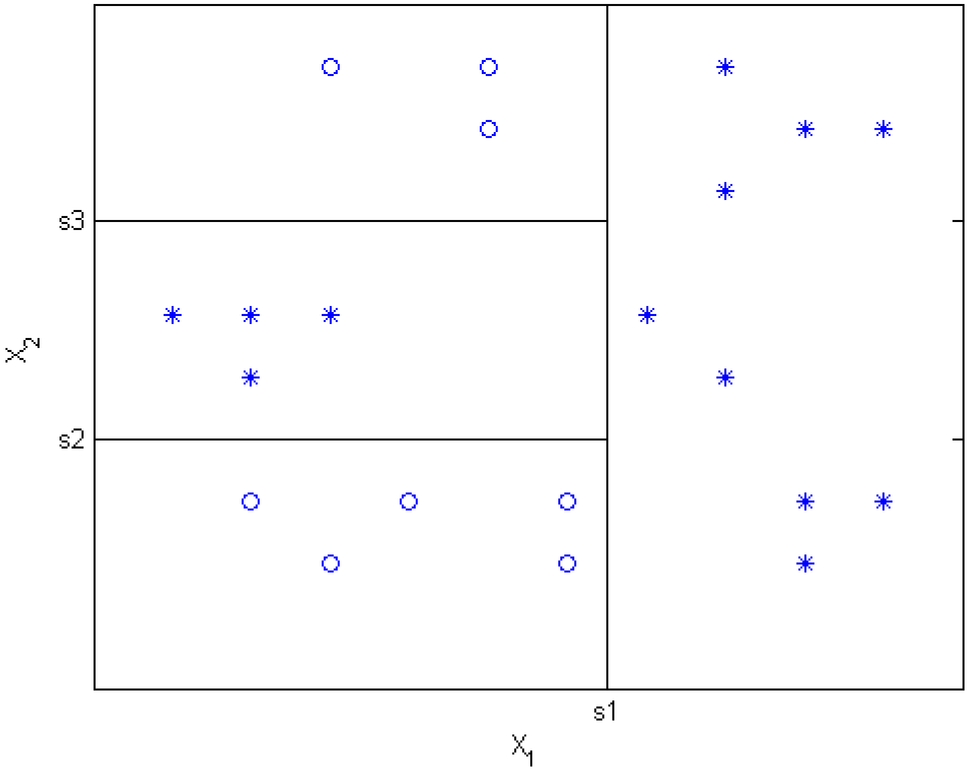


Figure 10. Recursive partitioning of two predictor variables for binary response

The recursive partitioning algorithm begins by selecting the feature variable X_a where $a \in \{1,2,\ldots,p\}$ that provides the best split of the data into M sub regions $\{R_1, R_2,\ldots,R_M\}$. The split is determined by finding the variable X_a and split point s that minimizes the sum of the impurity measure in each region for response y_i, and predicted response $\hat{y}_i$. The predicted response is the predicted classification of the observations in the region which is determined by the classification of the majority of the observations in the region. For a regression tree, we can use a squared-error impurity measure, but for classification, we require a different measure of node impurity. The Gini index and entropy measure are two popular choices of impurity metrics (Bell, 1996). We define the proportion of class k observations in region m as $\hat{p}_{mk}$, where N_m is the number of observations in region m, and I is the indicator function. It is this proportion that represents the predicted probability that an observation, or in this case a pixel, is classified as corroded or not corroded based on the values of the predictor variables, or features, at the pixel location.

$$\hat{p}_{mk} = \frac{1}{N_m} \sum_{x_{i \in R_m}} I(y_i = k)$$

$$\text{Gini} : \sum_{k=1}^{2} \hat{p}_{mk}(1 - \hat{p}_{mk}) \tag{12}$$

$$\text{Entropy} : \sum_{k=1}^{2} \hat{p}_{mk} \log(\hat{p}_{mk})$$

The partitioning process is continued until the minimum node size is reached. At this point the large tree, denoted T_0, is pruned using a method such as cost complexity pruning. T_0 is pruned by collapsing a number of the terminal nodes in T_0 to form a sub tree $T \subset T_0$. Cost complexity pruning uses an impurity measure, $Q_m(T)$, for sub tree T such as the Gini index or entropy measure, and a tuning parameter $\alpha \geq 0$ which controls the tradeoff between tree size and model accuracy. Given the number of terminal nodes in T, $|T|$, we define the cost complexity criterion $C_\alpha(T)$. For $\alpha = 0$, no penalty is given for increases in tree size, therefore T_0 would not be pruned. For each α, we find the sub tree $T_\alpha \subseteq T_0$ that minimizes $C_\alpha(T)$. It can be shown that there is a unique smallest sub tree, T_α, that minimizes $C_\alpha(T)$ (Gordon, 1999).

$$C_\alpha(T) = \sum_{m=1}^{|T|} N_m Q_m(T) + \alpha|T| \tag{13}$$

3.4. Data Processing

The data used in this research is processed from time-sequenced images of filiform corrosion growth in aluminum. Filiform corrosion occurs under the surface of coatings such as a paint primer. Corrosion filaments propagate in a worm-like fashion delaminating the protective coating. Our approach requires feature values for each pixel in time-sequenced images of corrosion growth. For non-corroded pixels, we use the pixel's feature values at time *t* to predict the possibility of corrosion at *t+1*. This approach requires a one-to-one mapping of pixels to ensure consistency from *t* to *t+1*.

3.4.1. Image Registration

Image registration is the process of aligning multiple images of the same scene. Typically, an input image is registered to align to a base reference image. Registration can be accomplished using the entire image area or the visible features in the image. Area-based registration uses correlation metrics, Fourier properties, or other image wide metrics to match multiple images. Feature registration is performed using points of reference that are visible in the images and determining an appropriate transformation to map the points in the input image to points in the base image. The images used in this research contain many visible landmarks which make feature registration an applicable method. Further definitions and details of image registration techniques used in this research are provided in Appendix A.

3.4.2. Image Segmentation

Once the image has been registered, the objects within the image need to be identified. In order to interpret an image, it must be decomposed into connected regions with similar characteristics. The process of image segmentation is to determine the appropriate partition D_i of the domain D of an image. Segmentation is actually a classification process, where pixels can be classified as belonging to an object, or to the image background (or not an object of interest). There are many appropriate techniques for segmenting an image. Two of the most popular methods are thresholding and edge detection (Neary, 2000). The images in this research were segmented using thresholding and edge detection techniques described in Appendix A.

CHAPTER 4. MODEL EVALUATION AND METRICS

This chapter discusses the properties of the proposed model that are influenced by the feature-based modeling approach. Specifically, the performance of the model is compared to a model without the feature information to lay the foundation for the empirical results in Chapters 6 and 7. In addition, the evaluation metrics used to compare the model performance are explained.

4.1. Properties of the Feature-Based Model

The feature based models for initiation and growth are dependent upon transition functions to determine the probability that a filament initiates at a given location, or the likelihood that a pixel transitions from an uncorroded state. Without the addition of the feature space variables, these functions are formed using information that is known about the underlying initiation or growth process. For example, we know that filament initiation sites occur along the defects in the coating material (scribe), therefore, locations outside of the scribe have a zero probability of initiation. In the absence of any other information, we could assign equal probability to sites along the scribe, or use a purely spatial "hot-spot" model to estimate the transition function. For the growth model, the known characteristics include the spatial dependency of growth so that new growth only occurs from previous growth, and that filament growth is self-avoiding so that only uncorroded locations are candidates for future growth. The inclusion of feature information allows the proposed models to provide a better approximation of the true underlying corrosion process.

The following sections detail the justification for the reduction in error for the feature-based models. In order to frame the problem we provide the following definitions.

Z: the set of all measurable features or attributes that can be utilized to describe the process.

W: the set of features and attributes that completely describe the process where $W \subseteq Z$.

V: the set of known characteristic features that any standard baseline model of the process must include. We assume these characteristics are necessary to completely describe the process, such that $V \subseteq W$.

4.1.1. *Lemma 1: Error Comparison - Classical Linear Model*

The total sum of squared error for the proposed model is less than or equal to the sum of squared error for the model without feature information.

Suppose $y = f(W)$ is the true transition function for the underlying process. Define $\hat{y}^{(1)} = h(V)$ as the approximation of the transition function using only the known baseline characteristics of the process. Let $\hat{y}^{(2)} = g(X)$ be the approximation of the transition function using the proposed feature-based approach, where $X \subseteq W$. The set X is assumed to include some or all of the features that describe the process in addition to the baseline characteristics, $V \subseteq X \subseteq W$. Therefore $\hat{y}^{(1)}$ includes only the baseline characteristics, while $\hat{y}^{(2)}$ includes the baseline characteristics and some or all of the additional features that describe the process. In the case of a general linear regression, where we assume the square of the matrix of predictor variables is positive definite and invertible, then the total sum of squared error for the proposed model (SSE[X]) is not greater than the total sum of squared error for the model without feature information (SSE[V]). The sum of squared error is defined as the sum of the difference in the true and predicted response over the number of observations, n.

$$\sum_{i=1}^{n}\left(y - \hat{y}^{(2)}\right)^2 \leq \sum_{i=1}^{n}\left(y - \hat{y}^{(1)}\right)^2 \tag{14}$$

The proof of the error comparison is built upon the assumption that $\mathrm{V} \subseteq \mathrm{X} \subseteq \mathrm{W}$ and follows similar derivations for the segmentation of residual terms (Draper and Smith, 1981) We denote the total sum of squares for the underlying process as SST which accounts for the sum of squared error due to the model (SSR) and the sum of squared error due to the pure error or randomness in the process (SSE). The definitions of the sum of squared error for the two models in the comparison are given below.

$$\begin{aligned} \mathrm{SST} = \mathrm{SSR}[\mathrm{V}] + \mathrm{SSE}[\mathrm{V}] \Rightarrow \mathrm{SSE}[\mathrm{V}] = \mathrm{SST} - \mathrm{SSR}[\mathrm{V}] \\ \mathrm{SST} = \mathrm{SSR}[\mathrm{X}] + \mathrm{SSE}[\mathrm{X}] \Rightarrow \mathrm{SSE}[\mathrm{X}] = \mathrm{SST} - \mathrm{SSR}[\mathrm{X}] \end{aligned} \tag{15}$$

If we define $\mathrm{SSR}[\mathrm{X} - \mathrm{V}]$ as the sum of squares for the model attributable only to the feature variables without the influence of the common baseline characteristic variables, then we can define the sum of squared error due to regression and error for the feature based model, $\mathrm{SSR}[\mathrm{X}]$ and $\mathrm{SSE}[\mathrm{X}]$.

$$\begin{aligned} \mathrm{SSR}[\mathrm{X}] = \mathrm{SSR}[\mathrm{V}] + \mathrm{SSR}[\mathrm{X} - \mathrm{V}] \\ \mathrm{SSE}[\mathrm{X}] = \mathrm{SST} - \mathrm{SSR}[\mathrm{V}] - \mathrm{SSR}[\mathrm{X} - \mathrm{V}] \end{aligned} \tag{16}$$

Because the terms involve the sum of squared terms, we are guaranteed that each term in the above equation is positive. We can define the relationship between $\mathrm{SSE}[\mathrm{X}]$ and $\mathrm{SSE}[\mathrm{V}]$.

$$\mathrm{SSE}[\mathrm{X}] = \mathrm{SST} - (\mathrm{SST} - \mathrm{SSE}[\mathrm{V}]) - \mathrm{SSR}[\mathrm{X} - \mathrm{V}] \Rightarrow \mathrm{SSE}[\mathrm{X}] = \mathrm{SSE}[\mathrm{V}] - \mathrm{SSR}[\mathrm{X} - \mathrm{V}] \tag{17}$$

Again, because all terms are positive, we are assured $\mathrm{SSE}[\mathrm{X}] \leq \mathrm{SSE}[\mathrm{V}]$, and can conclude the result in (14). Admittedly, the assumption that a classical linear model is an applicable estimate of the transition function is likely not valid for most processes. In fact, the output of the function is a density value in the case of initiation, and categorical in the case of growth. Therefore, the assumptions of normally distributed responses and error terms in the classical linear model are violated. The generalized linear model (GLM) seeks to remedy this by modeling a link linear relationship between the response and predictor variables. If we can assume that the response, y, is from the exponential family of distributions (Normal, Poisson, Binomial, etc), and $\mu = E[y]$, is related to a linear predictor, X through a monotone function, g, where $g(\mu) = X\beta$, then a GLM is appropriate. Because of their flexibility, GLMs have proven applicable to many applications (McCullagh and Nelder, 1989).

4.1.2. Lemma 2: Error Comparison - Generalized Linear Model

The scaled deviance of the proposed model is less than or equal to the scaled deviance for the model without feature information.

As with any function approximation, the parameters of the model are estimated by minimizing a goodness-of-fit criterion. In the case of the classical linear model with normal responses, we minimize the SSE. Parameters for GLMs are estimated using the method of maximum likelihood. If $f(y;\theta)$ is the density function for the observation y given the parameter θ, the log-likelihood is expressed as a function of the mean parameter μ and is based on independent observations $\mathbf{y} = \{y_1, y_2, \ldots y_n\}$. Given $\boldsymbol{\mu} = \{\mu_1, \mu_2, \ldots \mu_n\}$, we see that the density $f(y;\theta)$, is a function of the response for a fixed θ, as opposed to the log likelihood, $\ell(\boldsymbol{\mu}, \mathbf{y})$, which is function of θ, for a given set of data. The goodness-of-fit criterion for the GLM class of models is a function of the log likelihood classified as scaled deviance, $D^*(\mathbf{y};\boldsymbol{\mu}) = 2\ell(\mathbf{y};\mathbf{y}) - 2\ell(\boldsymbol{\mu};\mathbf{y})$, where $\ell(\mathbf{y};\mathbf{y})$ is the maximum likelihood attainable if the predictions perfectly matched the observations.

$$\ell(\boldsymbol{\mu}, \mathbf{y}) = \sum_{i=1}^{n} \log f(y_i, \theta_i) \tag{18}$$

Given the previous definitions of the sets of features included in the transition functions, we can extend the assertion that the feature based model of the transition function provides a better approximation than a model that ignores the features, for the case of the generalized linear model. If we define the scaled deviance of a proposed model using the feature set X as $D^*(\mathbf{y};\boldsymbol{\mu}_X)$, then $D^*(\mathbf{y};\boldsymbol{\mu}_X) \leq D^*(\mathbf{y};\boldsymbol{\mu}_V)$. As with the classical linear model, the proof of the result is based upon the nested models caused by the assumption that $V \subseteq X \subseteq W$. The proposed feature-based model accounts for the same predictors as the comparison model, but also includes other variables significant to the underlying process.

The deviance of the model, simplified as $D^*(\mathbf{y};\boldsymbol{\mu}) = -2\ell(\boldsymbol{\mu};\mathbf{y})$ since $\ell(\mathbf{y};\mathbf{y})$ is constant regardless of the model for comparison, acts as the SSR in the classical linear model. Minimizing the model deviance is equivalent to maximizing the log likelihood, $\ell(\boldsymbol{\mu},\mathbf{y})$. Therefore, the comparison of $D^*(\mathbf{y};\boldsymbol{\mu}_X)$ and $D^*(\mathbf{y};\boldsymbol{\mu}_V)$ is the complement of a comparison of the likelihoods. The parameter estimates for both models are determined through maximization of the likelihood functions, and the comparison of models results in a ratio of likelihoods, $\lambda = \ell(\boldsymbol{\mu}_V;\mathbf{y})/\ell(\boldsymbol{\mu}_X;\mathbf{y})$. It is shown through asymptotic likelihood ratio theory that $2\log(\lambda) \sim \chi^2$ with degrees of freedom $n-(p_X - p_V)$, where p_X and p_V are the number of parameters in the respective model (Shao, 2003). Because $2\log(\lambda) = D^*(\mathbf{y};\boldsymbol{\mu}_V) - D^*(\mathbf{y};\boldsymbol{\mu}_X)$ the same asymptotic distribution applies and therefore the difference in deviance is always positive and $D^*(\mathbf{y};\boldsymbol{\mu}_X) \leq D^*(\mathbf{y};\boldsymbol{\mu}_V)$. The assumption of the nesting of features implied by $V \subseteq X \subseteq W$, allows for a similar result for other model functional forms in applications when GLMs are not appropriate as in the case for tree models and other classifiers (Breiman, 1996b).

4.2. Model Evaluation Metrics

The previously described theoretic expectation is based on a goodness-of-fit metric to compare the model predicted response to the actual observed response. For the case of the classical linear model, we use sum of squared error, while for a GLM in classification, we use misclassification rate. Obviously, the goodness of fit metric used to fit the model should be considered when evaluating the model's performance, but other metrics provide additional insight into the model's predictive performance. The following sections provide a brief description of the metrics considered in the evaluation of the feature-based model for the macro and micro scale corrosion applications.

4.2.1. *Initiation*

The feature-based initiation model produces a density estimate over the region of interest. We are interested in determining if the estimated density improves our ability to predict the location of initiation sites. In the absence of feature information, a geographic-based hot spot model could be used as a baseline for comparison. However, the lack of obvious geographic clustering makes a uniform density a more realistic baseline model.

4.2.1.1. Likelihood

The likelihood function, $L(\theta)$, is the probability of occurrence of a sample of observations $\mathbf{y} = \{y_1, y_2, \ldots y_n\}$ given the probability density $f(y;\theta)$ with parameter θ is known, so that $L(\theta) = f(y_1)f(y_2)\cdots f(y_n)$. Therefore, we can calculate the likelihood of a sample of initiation site locations given the density estimate from the initiation model, $\hat{f}(y;\theta_X)$, where θ_X is used to represent the inclusion of features in the model. We estimate $\hat{f}(y;\theta_X)$ using a test set of data and then calculate the likelihood value for initiations in a test data set. We can then compare the likelihood values to a density estimate that does not include the feature information $\hat{f}(y;\theta_V)$. The baseline for comparison is for $\hat{f}(y;\theta_V)$ to be the uniform density. Therefore, we determine the likelihood of the initiation sites given the feature based density as opposed to a random selection of locations.

The use of simulation allows us to make inference as to the significance of the likelihood results. Samples can be drawn from the estimated density, $\hat{f}(y;\theta_X)$, in order to produce a distribution of likelihood values assuming $f(y;\theta) = \hat{f}(y;\theta_X)$. This distribution is then used to determine the significance of the observed likelihood. In essence, we calculate a p-value, which is a measure of confidence that the observed likelihood comes from the estimated density function $\hat{f}(y;\theta_X)$. In similar fashion, we can draw samples from the baseline density $\hat{f}(y;\theta_V)$ and produce a distribution of likelihood values assuming $f(y;\theta)=\hat{f}(y;\theta_X)$. This is the distribution of expected likelihood values if the estimated density was incorrect and the process was truly random, or not dependent on the included features. The likelihood distributions allow us to perform two different hypothesis tests. The first distribution is used to test the null hypothesis that the estimated density is correct, while the second distribution is used to test the null hypothesis that the random model is correct.

4.2.1.2. Percentile Score

Percentile score provides another metric for evaluating the predictive ability of a proposed model with specific applicability to point processes over a geographic region. Percentile score provides a quantitative metric for comparing the density estimate at the observed location to the density estimates at all other locations, thus providing a measure similar to the likelihood of the observation. Percentile scores have been used to evaluate models for crime and terrorism where the modeled events are the locations and times of occurrence (Liu and Brown, 2003; Smith, 2005).

Given that the pixels in the images form the vertex set $\mathcal{V}$ with specific location $i \in \mathcal{V}$, where $|\mathcal{V}| = I$, we wish to calculate the percentile score at the location of an event, s, denoted as $\pi(s)$. The definition utilizes the indicator function, $\mathbf{1}(a \le b)$, which evaluates to 1 if $a \le b$ and 0 otherwise.

$$\pi(s) = \frac{1}{I}\sum_{i \neq s} \mathbf{1}\left\{\hat{f}(s;\theta_X) \ge \hat{f}(i;\theta_X)\right\} \tag{19}$$

In the case of multiple initiation sites $S = \{s_1, s_2, \ldots, s_n\}$, we can average the percentile score over all initiation sites to determine $E[\pi(s) \mid s \in S]$. The resolution of the vertex set determines the ability of the percentile score to approximate the relative magnitude of the density estimates.

4.2.2. Growth

The complex nature of growth processes make them very difficult to model and predict. Because of the dependence between successive configurations, an error from the interacting particle model at a given point in time propagates into future predictions. The complexity of the underlying process makes it difficult to determine an accurate metric for model evaluation. Most researchers focus on general characteristics of the object being modeled. For example, a great deal of research for interacting particle models can be found with application in ecology, including models for the spread of wild fire (Karafyllidus and Thanailakis, 1997; Liu and Chou, 1997; Li and Magill, 2001; Berjak and Hearne, 2002; Preisler *et al.*, 2004; Sullivan and Knight, 2004). The predicted fire spread is compared to actual fire spread data, but typically only qualitative metrics are used to evaluate the predicted fire spread. The ability of the model to produce concentric rings of fire spread at successive time periods and the effect of changes in terrain and wind observed in the prediction is compared to actual data to ensure similarity in the results. Although the ability of the model to mimic the general characteristics of filiform growth is fundamental, we utilize more quantitative metrics in order to determine the significance of the difference in performance of the feature-based and random models.

4.2.2.1. Recall and Precision

If we assume the realization of the interacting particle model produces a predicted filament over the given time period, then we can compare the filament to the actual filament using the classification of the individual pixels in the image. Classification metrics are based on values calculated in a confusion matrix. Table 2 illustrates the construction of a confusion matrix as the number of true negative (TN), false negative (FN), false positive (FP), and true positive (TP) predictions.

		Truth	
		0	1
Predicted	0	TN	FN
	1	FP	TP

Table 2. Confusion matrix components

Two popular metrics for classification models are sensitivity and specificity. Higher sensitivity indicates fewer positive responses going undetected. High specificity indicates fewer misclassified negative responses. Information retrieval systems are evaluated using two similar metrics: precision and recall. Information retrieval systems, such as web search engines, seek to classify information as being relevant to some specified query. Precision is the proportion of relevant information out of all the information retrieved. Recall is the proportion of relevant information out of all the available relevant information. The recall metric is identical to the sensitivity metric; however there is a subtle difference in the calculation of precision and specificity. Precision is a measure of the positive predictive power of the model, given that the denominator covers the predicted positive response. Although either set of metrics can be considered applicable to corrosion, we chose the use of precision and recall due to the higher emphasis on positive predicted and actual responses.

$$
\begin{aligned}
\text{Sensitivity} &= \frac{\text{TP}}{\text{TP} + \text{FN}} \\
\text{Specificity} &= \frac{\text{TN}}{\text{TN} + \text{FP}} \\
\text{Pr ecision} &= \frac{\text{TP}}{\text{TP} + \text{FP}} \\
\text{Re call} &= \frac{\text{TP}}{\text{TP} + \text{FN}}
\end{aligned}
\tag{20}
$$

Precision and recall have a logical translation in terms of probability. Precision can be defined as the probability that the object is truly positive, given that the classifier predicted positive. In the case of corrosion, precision is the probability that a pixel corrodes given that the model predicted that it would. Recall represents the probability that a positive is correctly classified, or that a corroded pixel is accurately classified.

4.2.2.2. F-Measure

Typically, a tradeoff is required between precision and recall because as we are able to capture more of the true positive response (higher recall), our precision of predicted positives goes down (lower precision). Precision and recall curves can be used to determine the most appropriate outcome in terms of both metrics. The F-measure offers a way of combining the two metrics that is more sensitive to differences than just taking the arithmetic mean. The F-measure is the harmonic mean between precision and recall, which is appropriate when an average of rates is required. The F-measure surface shown in Figure 11 illustrates the tradeoff between precision and recall and the emphasis placed on balancing values of both metrics.

$$\mathrm{F} = \frac{2 \cdot \mathrm{Precision} \cdot \mathrm{Recall}}{(\mathrm{Recall} + \mathrm{Precision})} \tag{21}$$

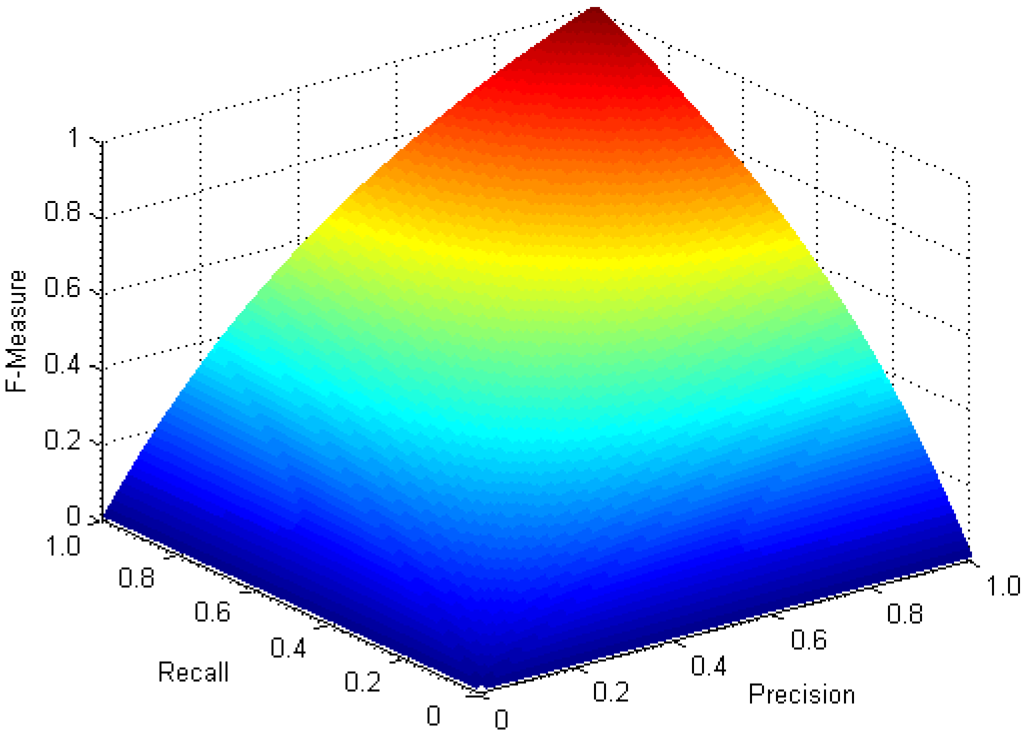


Figure 11. F-measure metric for classification

CHAPTER 5. APPROACH

This chapter describes the approach for two scales of data used in the application of the proposed models. First, the macro scale approach is described including the available data and derived features, followed by a description of the micro scale approach. Further detail is provided for the data and features used at the micro scale due to the significance of the inclusions as features for the model.

5.1. Macro Scale

The initial approach of the previously described methodology involved the use of filiform corrosion data at a macro scale. Preliminary data were obtained for the growth of filiform corrosion on three samples of aluminum AA2024-T3 by Daryl Little, PhD candidate in the Department of Material Science and Engineering at the University of Virginia. Each 50mm x 50mm sample was polished to 1200 grit, coated with epoxy polyamide (similar to an aircraft primer), scribed, and exposed to concentrated HCl for 30 seconds. The samples were kept at 40°C and 80% relative humidity. After one day of exposure, images of the sample were taken using a digital camera. The images were 2048 x 1536 pixels covering an area of 19.05 mm x 25.4 mm, for a resolution of 6501 $\text{pixels}/\text{mm}^2$. The images were grayscale with intensity level [0,255]. Images were cropped to capture the scribe and filiform growth. Data was derived from the images of the filiform growth from day 1 to day 2. Images from samples 1 and 2 were used for training data, and sample 3 was kept as a test set.

5.1.1. Data Images

The macro scale images were not originally produced for this research, and therefore the resolution and consistency of the images made it difficult to extract the data. Each pixel in the image represents a single observation, and the value of the pixel can be classified as either corroded or not. Because no other data was recorded for the samples, the only features available for consideration were those that could be directly derived from the images. Image registration was performed using the landmarks visible in the coating that were distinguishable in each image. Figure 12 and Figure 13 show the time sequenced images for the two training samples. The filaments cannot be individually distinguished, but the general shape of the filiform corrosion can be seen with active heads visible as the dark tips of corrosion objects.

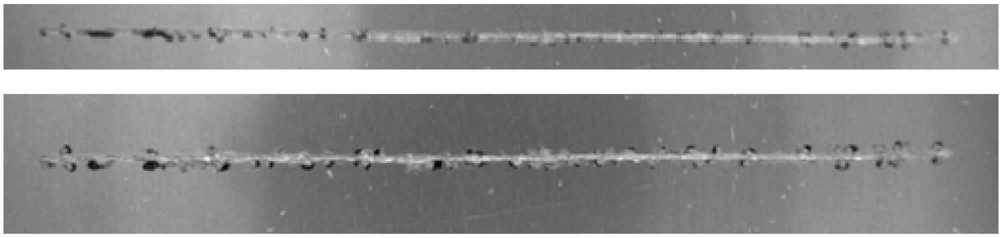


Figure 12. Sample 1 images for day1 (top) and day 2 (bottom)

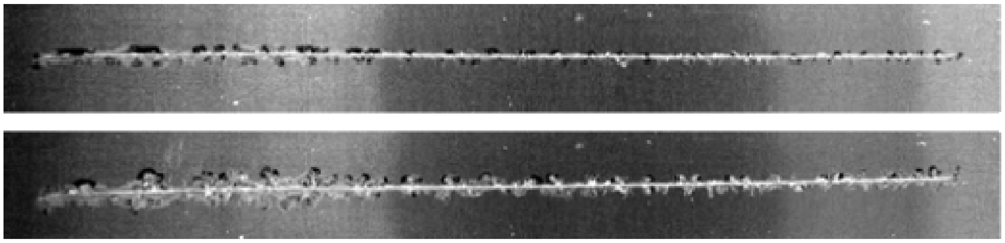


Figure 13. Sample 2 images for day1 (top) and day 2 (bottom)

The quality of the images made the automated image segmentation techniques unreliable. Therefore, due to the small number of images available, segmentation was performed by hand. By zooming into the corrosion and scribe areas, the pixels were manually classified as corrosion and scribe. Figure 14 shows the result of the manual segmentation on the day 1 and day 2 images for sample 3. The corroded pixels are visible in green and the pixels that make up the scribe are visible in blue.

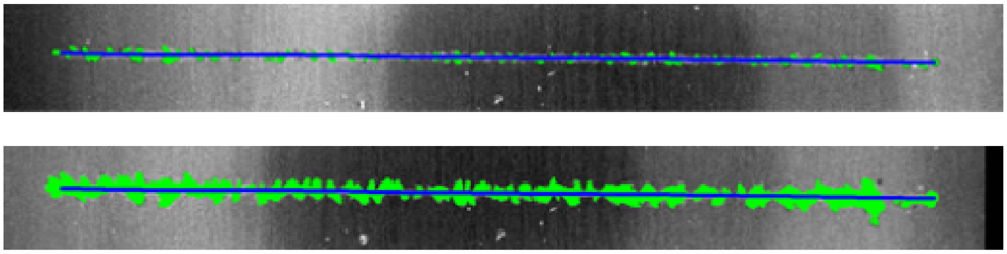


Figure 14. Sample 3 images for day 1 (top) and day 2 (bottom) with segmented scribe (blue) and corrosion (green)

5.1.2. *Derived Features*

The image processing produces a classification for each pixel which becomes the response data for modeling. After image processing, the feature values for each observation were derived. Each group of corroded pixels is considered a corrosion object, and new corrosion objects initiate from a single point along the scribe. The initiation point for each object was determined using centroid data for each object distinguished in the day 1 image. The most obvious features are the distance from each pixel to the scribe, and the distance to the nearest initiation point. In general, we would expect pixels closer to the scribe and/or to an initiation point to be more likely to corrode than those that are farther away. Figure 15 depicts the calculation of the distance to the scribe (a=6) and the distance to the closest initiation point (b=11.8) for a given pixel in the test data set: [row, column]=[128,197]. Dark pixels represent uncorroded pixels, and the scribe is the white area in the bottom of the image. The metrics are given for the pixel of interest shown as the single white square in the image. At t=0, no corrosion is present, so the available features for the prediction of growth to t=1 is limited to the above distance metrics for this scale approach.

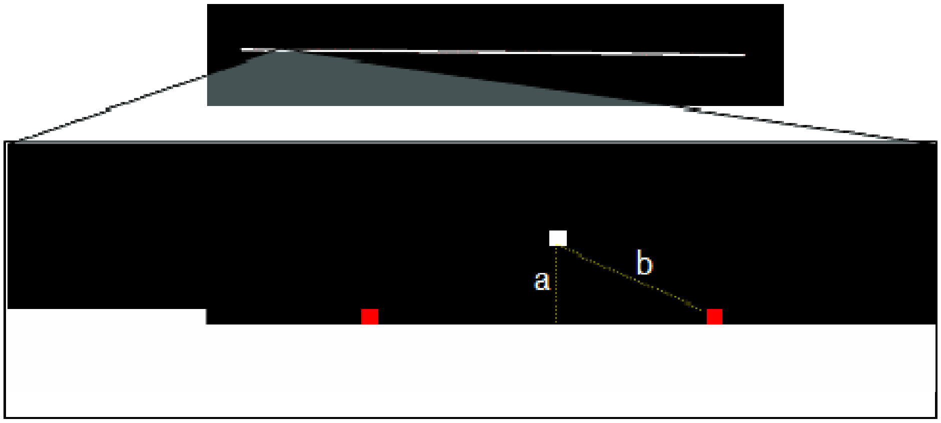


Figure 15. Distance to scribe (a) and distance to nearest initiation point (b)

Other features can be generated for predicting growth for t>1. Figure 16 illustrates some of the geometric features that can be derived using the corrosion objects visible in the images. The angle to the nearest initiation point (c=2.68), captures the propensity for filaments to grow away from the scribe. The distance to the nearest corroded pixel (d=7.81) accounts for the spatial dependency of new growth only occurring from current corrosion. The distance left to a corroded pixel (e=9), distance right to a corroded pixel (f=14), and the resulting gap distance (g=23) were used to determine if pixels that fall between corrosion objects are likely to corrode and therefore cause the objects to grow together.

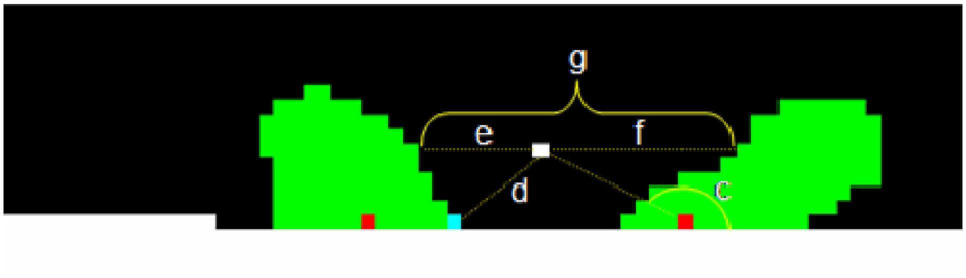


Figure 16. Angle and gap feature calculations

The amount of corrosion near a pixel should influence the likelihood for that pixel to become corroded. The following figure illustrates the variables derived to capture this relationship. The number of corroded pixels within a 3x3 region (h=0), 5x5 region (i=0), 9x9 region (j=0), and 17x17 region (k=13) around the pixel are in Figure 17.

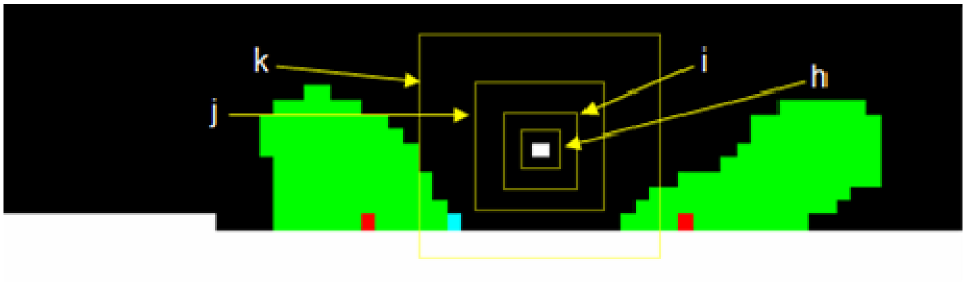


Figure 17. Features for the amount of corrosion in surrounding regions

The angle from the nearest corroded point to the pixel (l=0.69), the angle from the initiation point to the nearest corroded pixel (m=0), and the distance from the initiation point to the nearest corroded pixel (n=6) are illustrated in Figure 18. Additional features were derived and calculated from the features explained above. The size and solidity of the nearest corroded object were considered, as well as the extent of the region around the nearest corroded pixel. Functional variables included the product of gap width and distance to scribe and a function of the angles and distances for nearest corroded pixel and initiation point, *dang_ratio*. The variable *dang_ratio* was derived to capture the propensity of the filaments to continue growth in the direction of the current growth. The feature *dang_ratio* is a function of the features *pc_ang*, as the angle from the pixel to the nearest corroded point, *ci_ang* as the angle from the initiation point to the nearest corroded pixel, *corr_d* as the distance to the nearest corroded pixel, and *ci_d* as the distance from the initiation point to the nearest corroded pixel. Including all of the derived variables, created a data set of 19 features and over 400,000 observations.

$$dang\_ratio = \left|pc\_ang - ci\_ang\right| \frac{corr\_d}{1+ci\_d} + \frac{corr\_d}{1+ci\_d} \tag{22}$$

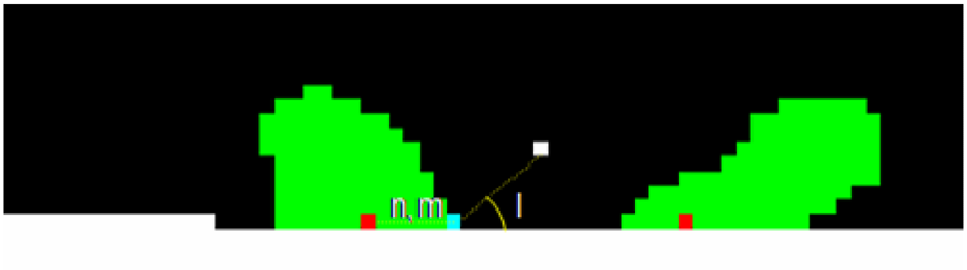


Figure 18. Other distance and angle features

5.1.3. Analysis of Feature Data

The initial analysis of the derived feature data provided some useful insights into the process. The histograms in Figure 19 compare the distribution of selected feature values for corroded and non-corroded pixels at t=2 for the training data. The majority of corrosion at t=2 occurs within a distance of 35 from the scribe. Therefore, we can eliminate pixels well beyond that range from the model datasets and considerably reduce the number of observations. Roughly half of the corrosion occurs where small gaps exist between current corrosion, but the other half occurs where no gap exists. The angle features are more difficult to analyze individually because pixels that are farther away do not corrode and have a tendency to be at an angle close to π/2. Corroded pixels are also more prevalent at π/2, but are fairly uniform in the tails. It is difficult to draw many classification conclusions from the histogram data. The lack of an obvious linear discriminant in any of the features makes it difficult to identify specific distinguishing features.

The correlation matrix provided in Appendix B, highlights the strength of the relationship between features and the response. There are several features that are highly correlated, which is expected given the manner in which they are calculated. For example, we expect the left and right distances to be highly correlated with the overall gap distance. The gap features as well as the number of corroded pixels in the various regions around the specified pixel are significantly correlated with the response variable.

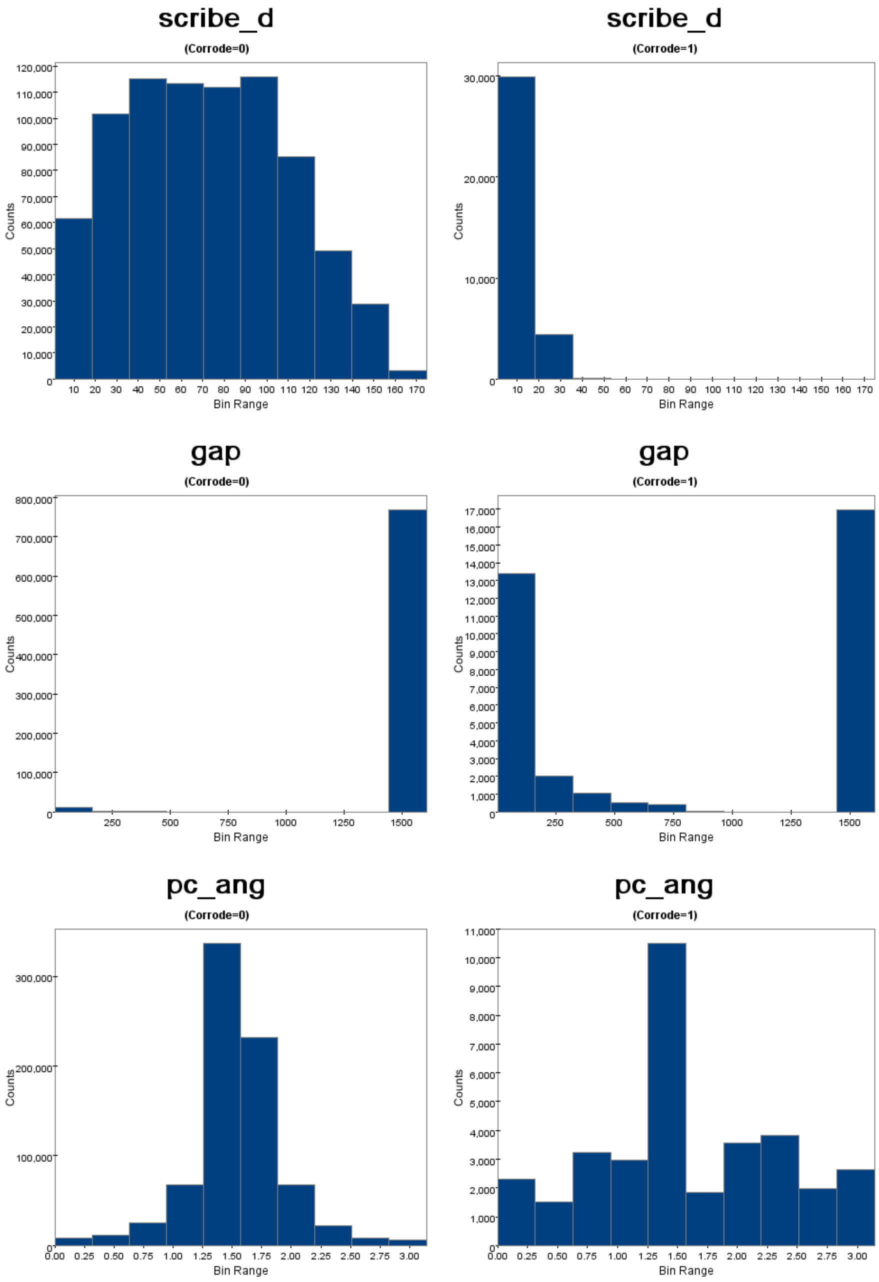


Figure 19. Histograms of selected feature data (non-corroded vs. corroded)

5.2. Micro Scale

The micro scale approach uses filiform growth data on samples similar to the macro scale data. The desire to include features inherent to the underlying microstructure of the material made it necessary to generate samples specific to this research. This allowed for more standardized images at a much higher resolution so that the microstructure features could be discerned. Multiple samples (50mm x 50mm) of AA2024-T3 were prepared in almost identical fashion to the macro scale samples by Connor Parker, MS student in the Department of Material Science and Engineering at the University of Virginia. The only significant difference was the polishing of the surface prior to coating. In the macro scale experiment the samples were polished to 1200 grit, but a higher degree of polishing was required for the micro scale data in order to make the microstructure features visible. The samples were polished to a 1μm finish, coated with epoxy polyamide, scribed, exposed to concentrated HCl for 30 seconds, and stored at 40°C and 80% relative humidity.

5.2.1. *Image Processing*

The samples were imaged using a confocal laser scanning microscope (CLSM) at 20X magnification prior to HCl treatment, $t = 0$, to provide a baseline image for comparison to subsequent growth images. Once exposed, images were taken at $t = \{6,21,28,44,51,67,73,89\}$ hours. The magnification of the images in the CLSM greatly changed the scale of the images. The motorized XY scanning stage ensures that the same area of the sample can be imaged at consecutive time periods. A template was also used to hold the samples in the same location and orientation on the stage. The template and motorized stage standardized the imaging process and greatly simplified the image registration process.

Instead of a single image per sample, 13 images were required to cover the entire sample width along the scribe. Therefore, the sample was partitioned into 13 areas with an image taken of each area at each time period. Areas 1 and 13 were the edges of the sample which did not include any scribe length, therefore these areas were not included. Additionally, the tips of the scribe were captured in areas 2 and 12. In order to maintain consistency between images, the tips of the scribe were ignored, therefore, the modeling and analysis in this research was performed using images from areas 3-11. Each image was 4096x2048 pixels covering an area of 4x2mm, for a resolution of approximately $1 \text{ pixel}/\mu\text{m}^2$.

Figure 20 shows time sequenced images of area 7 in sample 1 for time periods $t = \{6,28,51,73\}$ hours. The complete time sequenced images for all 9 areas can be seen in Appendix H. The square patterns visible in the images are due to the stitching process that the CLSM uses to scan large areas. Each square area is approximately 0.25 mm^2. The scribe is the dark horizontal line through the middle of each image. At $t = 6$ hours, several small dark bubbles are visible along the scribe. These bubbles represent points of initiation and are the beginnings of filament heads. The remaining images show how the filaments evolve. The filament on the left side of the image grows out perpendicular to the scribe, while the filaments in the middle of the image appear to take a more geographically random path.

The ability to distinguish individual filaments in the images improved the ability to segment the image into objects. The previously described image segmentation techniques were applied to each image. The primary objects to identify were the scribe and the filaments at various time stages. Because the images were registered, the scribe and other visible heterogeneities only needed to be located in the $t = 0$ image. Once these objects were identified the relative filament position can be determined regardless of the time period.

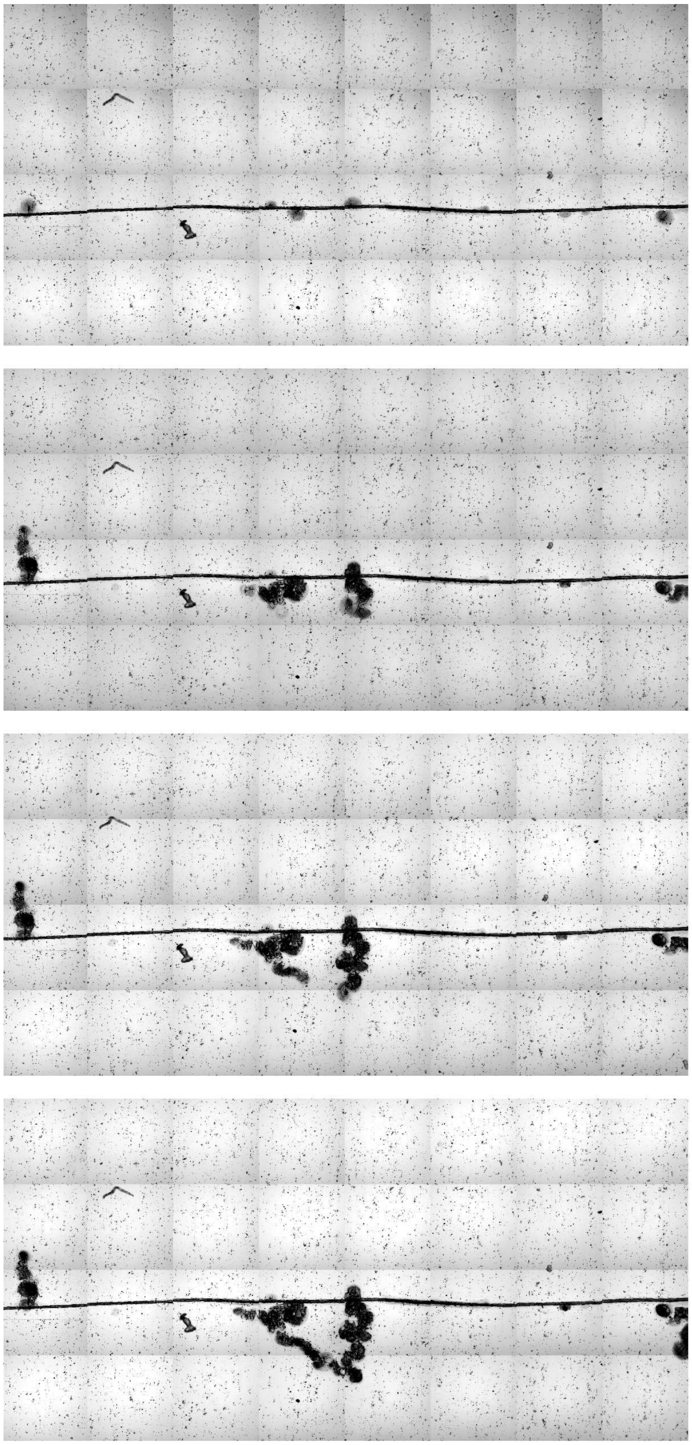


Figure 20. Area 7 images: t=6, 28, 51, and 73 hours (top to bottom)

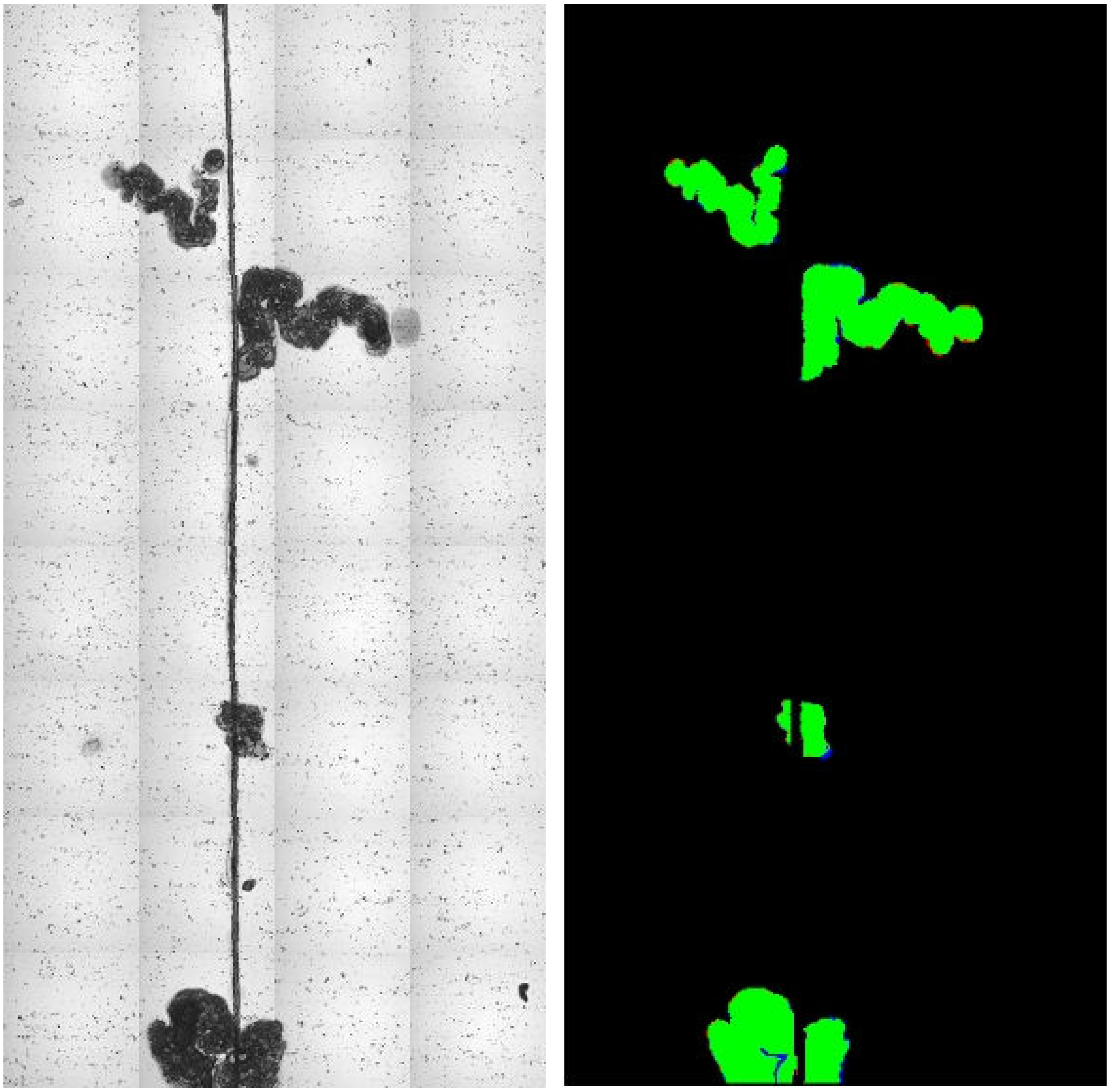


Figure 21. Accuracy of image segmentation

Although the contrast between the filaments and the background of the image is fairly distinct, tight turns in the filament path, changes in pixel intensity, and heterogeneities in the image can make segmentation of the filament growth difficult. Despite the difficulties, the thresholding and edge detection techniques detailed in Appendix A, proved to be very effective at capturing the filament growth. In order to validate the segmentation accuracy, one of the nine areas in sample 1 was segmented by hand to approximate a truth baseline for comparison. The segmentation methods were then applied to the area and the results were compared to the truth classification.

Figure 21 shows the result of the validation of segmentation accuracy. The original image is shown on the left while the results of the comparison of the segmentation methodology to manual segmentation are shown on the right. The pixels in green are accurately classified as part of the filament. The red pixels are locations that were false negatives (i.e. should have been classified as filament), and the blue pixels represent false positive classifications (i.e. should not have been classified as filament). The confusion matrix of the segmentation accuracy results are shown in Table 3. The segmentation produced a type I error rate of 1.53% and a type II error rate of 0.16%. The overall misclassification rate of the entire area was 0.22%. The type II and overall misclassification rates are somewhat skewed by the large area not included in the filament growth. The ability to classify pixels as not belonging to the filament becomes less important as the distance from the true filament increases. However, the type I rate is the primary concern given the desire to predict the locations of future filament growth. Therefore, given the low rate of type I error, we conclude that the performance of the segmentation methodology is satisfactory.

		Truth		
		0	1	Total
Predicted	0	7,990,745	12,654	8,003,399
	1	5,900	379,309	385,209
	Total	7,996,645	391,963	8,388,608

Table 3. Confusion matrix for segmentation accuracy

Each micro image was registered and segmented in order to produce a precise pixel by pixel mapping of all 9 areas at each of the 9 time periods. The scribe, heterogeneities and filament growth was segmented in each image, so that in any area and at any time period, each pixel can be classified as corroded or not. The identification of the heterogeneities also allows for these objects to be used as the features in classification models similar to the macro scale images. However, the dramatic improvement in resolution and standardization between images allows for significant improvements in data accuracy, and the ability to model individual filament growth.

5.2.2. Analysis of Filament Growth

The accurate segmentation of individual filaments allows for a more detailed analysis of filament growth characteristics. Of particular interest is the rate of growth and the size, or spread, of the filament path. The growth model described in Chapter 3 utilizes an interacting particle methodology to simulate the growth of a single filament. The empirical data for filament growth in the CLSM images can be used to control the growth rate and filament width as determined by the local transition function, f, and the growth probability function, p.

5.2.2.1. Filament Growth Rate

Analysis of individual filament growth was performed on a subset of filaments from sample 1. A total of 10 filaments were chosen from various areas of sample 1. The filaments were selected based on the ability to crop out the filament from the image. Filaments near the border of the image were not selected to eliminate any edge effects and loss of growth data. Filaments in close proximity to other filaments were not selected in order to ensure the accuracy of the growth data.

The growth rate for individual filaments can be determined using the segmented growth from the time sequenced images. The growth rate for a filament from time $t = 6$ to $t = 21$ can be determined by taking the growth at time $t = 21$ and subtracting off the growth at $t = 6$. The rate of growth is defined as the number of pixels added to the filament divided by the length of time. Growth rates for the filament are shown in Figure 22. The rates at time $t = 6$ are inconsistent because the exact time of initiation is unknown and variable from filament to filament. For example, if we look at the image at $t = 6$, a filament that initiated in the first hour will show more growth than a filament that initiated in hour 5. The rates for the other time periods appear varied, but clustered within a banded region. Figure 23 shows the mean growth rate across all filaments. The error bars represent +/- one standard deviation from the mean, and the dotted line shows the mean across all time periods. Except for the first time period, the mean falls within the one standard deviation range over all time periods, which suggests that a constant growth rate could be applicable. In fact, a linear regression of the mean growth rate returns an intercept term near the mean growth rate value and the coefficient for the time period variable is not significantly different from zero at any confidence level. Therefore, a mean constant growth rate for the interacting particle model is a reasonable assumption.

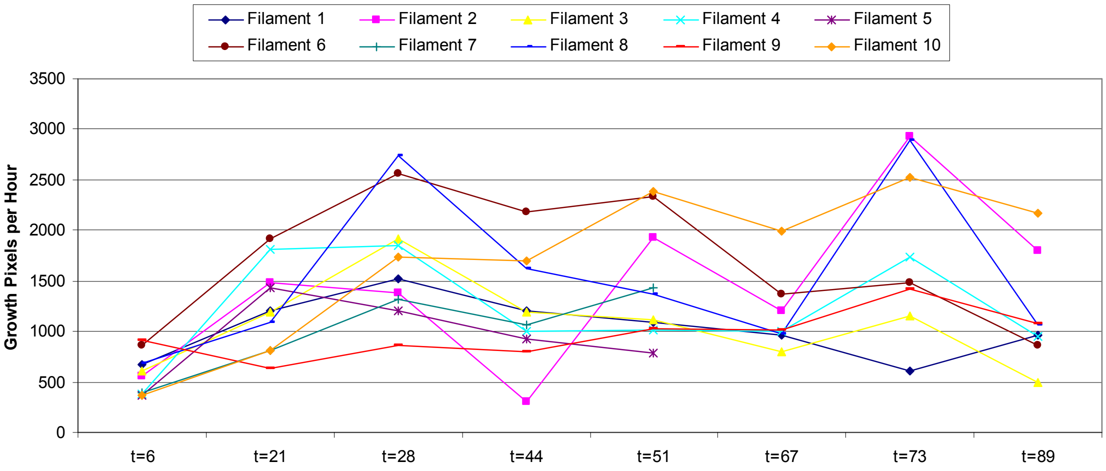


Figure 22. Growth rates for sample 1 filaments

Although the rate appears constant, the data appears to exhibit a cyclical up and down pattern which could suggest that the growth rates at subsequent time periods are correlated. Autocorrelation functions (ACF) provide a method for calculating the correlation between time series data (Shumway and Stoffer, 2005). The ACF can be useful in determining if the data is random, if adjacent observations are related, if the data is sinusoidal, and if the time series is autoregressive. The sample ACF is denoted ρ_h, for observed values $\{Y_1, Y_2, \ldots, Y_N\}$ at times $\{t_1, t_2, \ldots, t_N\}$, where h is the specified lag period. The ACF is a correlation coefficient which measures the relationship between observations of the same variable at times t_i and t_{i+h}.

$$\rho_h = \frac{\sum_{i=1}^{N-h} \left(Y_i - \overline{Y}\right)\left(Y_{i+h} - \overline{Y}\right)}{\sum_{i=1}^{N} \left(Y_i - \overline{Y}\right)^2} \qquad (23)$$

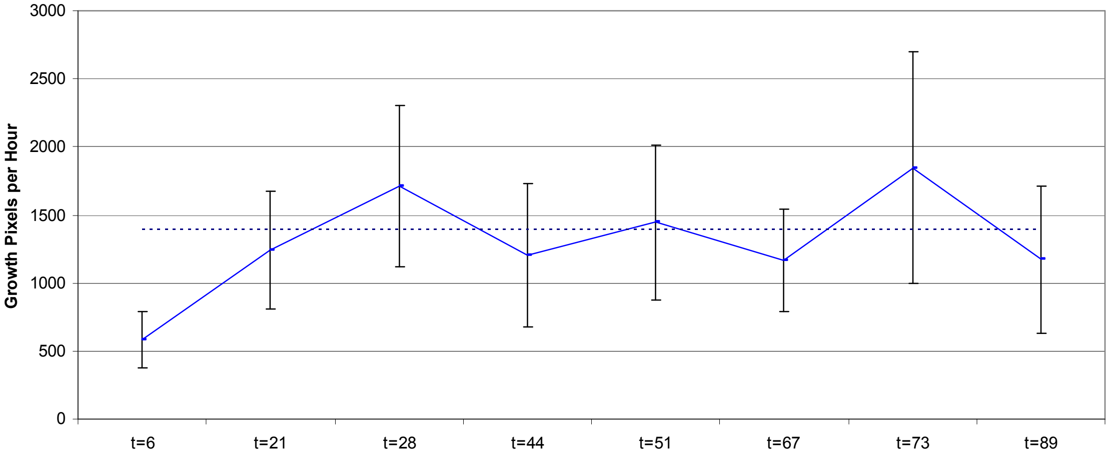


Figure 23. Mean growth rate for sample 1 filaments

Plots of the ACF function are typically used to analyze time series data. ACF values near zero imply the lack of time dependence in the data. The ACFs for the mean growth rates are provided in Figure 24. All of the ACF values are close to zero and well within the 95% confidence bounds shown as dashed lines on the figure. The distribution of the sample ACF is approximately normal for large N, therefore significant ACF values will fall outside of the interval $\pm z_{\alpha/2}/\sqrt{N}$, where $z_{\alpha/2}$ is the standard normal variable where $P(|z| > z_{\alpha/2}) = \alpha$ (Shumway and Stoffer, 2005). The confidence bounds in Figure 24 were calculated under the large sample assumption; however, the low ACF values do not suggest time dependence in the rate data.

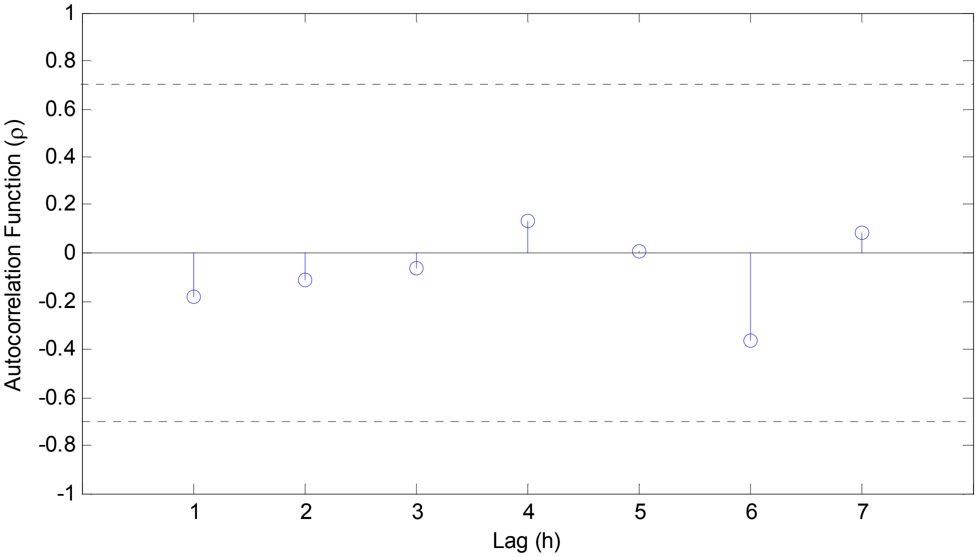


Figure 24. Autocorrelation plot for filament growth rate

The partial autocorrelation functions (PACF) can also be used to test for time dependence in data. The PACF determines the amount of correlation between an observation and its lagged observation that is not explained by lower order lags. For example, if the observations show correlation at lag 1, then the ACF at lag 2 could also be significant due to the impact of the lag 1 correlation. The PACF removes the correlation from the lower lag. The PACF for the growth rate data is shown in Figure 25, and the results are similar to the ACF results. The growth rate data does not appear to exhibit any significant time dependence, therefore a constant mean rate of growth is a reasonable assumption.

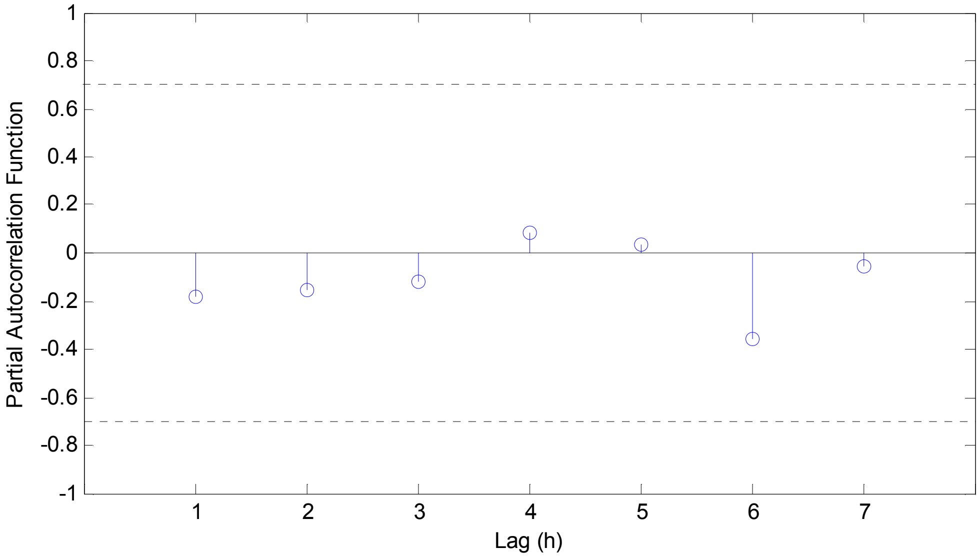


Figure 25. Partial autocorrelation function for filament growth rate

5.2.2.2. Filament Width

The example filament simulation given in Chapter 3 uses a variable *Spread* to control the width of the object. As the filament head moves forward, the locations along the path of the head continue to corrode outward as long as the chemical reaction can be maintained. Observation of the filaments in the CLSM images suggests that the width of the filaments is generally consistent. Slight width variations can be seen from the initiation point to the head location, and the width typically tapers to the head due to the amount of time that a location has been susceptible to corrosion. As a generalization, however, the width appears constant and consistent from filament to filament.

The width of the filament is another important consideration for modeling the local transition function, f, and the growth probability function, p. We can use empirical information from the images to create a distribution of filament widths which can be used in the growth model to control the amount of time that corrosion continues after the head of the filament passes over a location. The data for filament width was calculated using the filaments selected for the growth rate analysis. The width was determined using the "pixval" function in the Image Processing Toolbox for Matlab 7.0. Width measurements were taken in three different locations along the filament path: near the initiation point, in the middle of the path length, and at the end near the filament head. Three different measurements were taken at each location for a total of 90 measurements.

The distribution of the width measurements is shown in Figure 26. The width measures range from 60 to 170 pixels. Closer inspection of the width distributions supports the previous assertions of filament growth. Figure 27 shows the width distribution grouped by the location along the filament path. Three modes are visible in the data. Areas near the initiation point are generally wider than areas closer to the head because of the length of time that these areas were susceptible to corrosion.

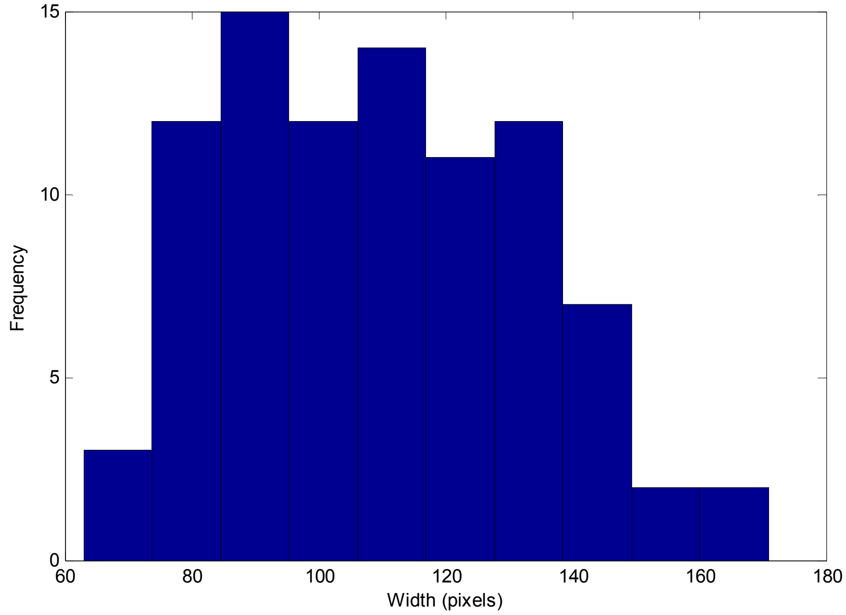


Figure 26. Distribution of filament widths

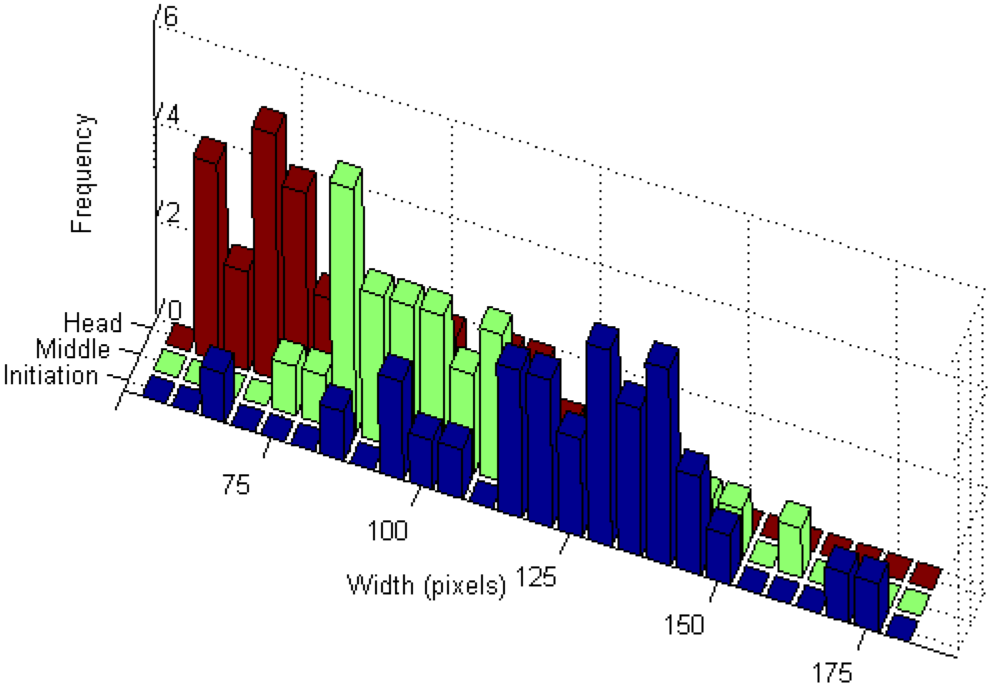


Figure 27. Distribution of filament width by path location

5.2.3. Determination of Data Points

The increase in resolution from macro to micro scale creates an enormous increase in the amount of data available for modeling at this scale. A single image was used at the macro scale for each time period, resulting in a training data set of 400,000 observations. A single image of 1 area from the CLSM contains over 8 million pixels. Considering there are 9 areas per sample, the amount of data available quickly becomes unmanageable: 72,000,000 pixels/sample x 4 samples x 8 time periods = 2.3 billion pixels.

The purpose for generating these data sets is to use the classification methods discussed in Chapter 3 to estimate the growth probability function, p, defined in the growth model. The function maps the feature values into the [0,1] probability line. In order to model the probability that a pixel corrodes, or becomes part of the filament, we need to determine which pixels are actually candidates for corrosion. The filament head proceeds under the material coating and areas along the path become the corroded filament. Therefore, only pixels near the head of the filament at time t are truly candidates for corrosion at $t+1$.

At each time period, the location of the head of the filament is known and can be identified at the end of the path. Therefore, the actual corrosion candidate pixels are those in the area around the filament head. Figure 28 illustrates the time sequenced growth of a filament. The filament grows from the initiation point along the scribe on the left side to the tip of the filament on the right. The shades of grey represent the growth at successive time periods. In general, the border pixels represent the head of the filament at a given time period. However, the growth in filament width after the head passes through a location causes some of the narrow wrap around areas visible along the borders in the figure. The head pixels at a given time period can be easily isolated by dilating the growth at time $t+1$ and taking only those pixels included in the growth at time t. The wrap around pixels caused by the width growth can be eliminated using a morphological opening of the growth at time $t+1$ which erodes away the narrow portions of growth leaving only the wider path of the filament head.

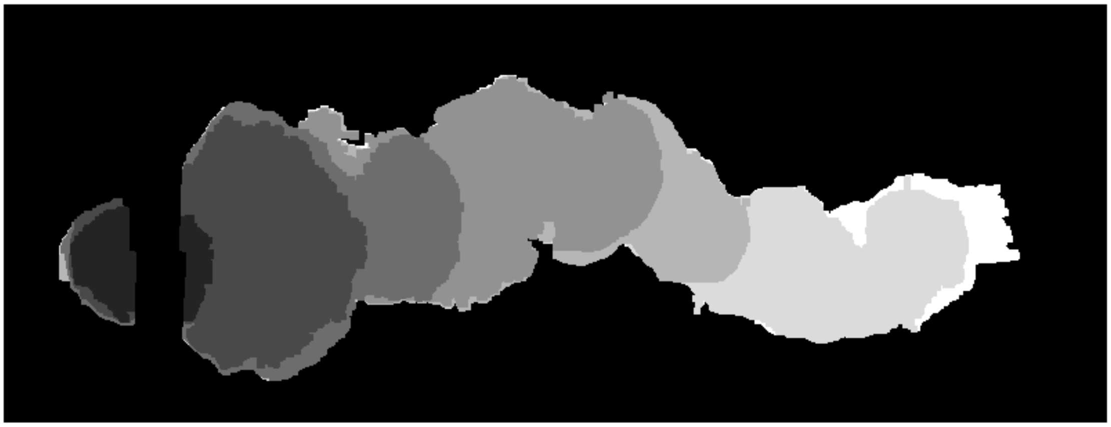


Figure 28. Time sequenced growth of a single filament

Once the pixels for the filament head are identified, the candidate pixels are determined as those pixels in the region around the head. The resulting areas represent the frontiers of possible growth at the next time step. Figure 29 shows the growth frontiers for the filament in Figure 28. The white pixels show those locations that did not corrode at the next time step, and the grey pixels represent locations in the path of the filament head. The growth frontiers were calculated for the filaments in each of the nine areas of sample 1. Identifying the data points using the growth frontiers produced over 2.1 million observations, which is significantly less than the total number of pixels in the images. More importantly, the frontiers provide a much more accurate representation of the true candidates for corrosion at each time period.

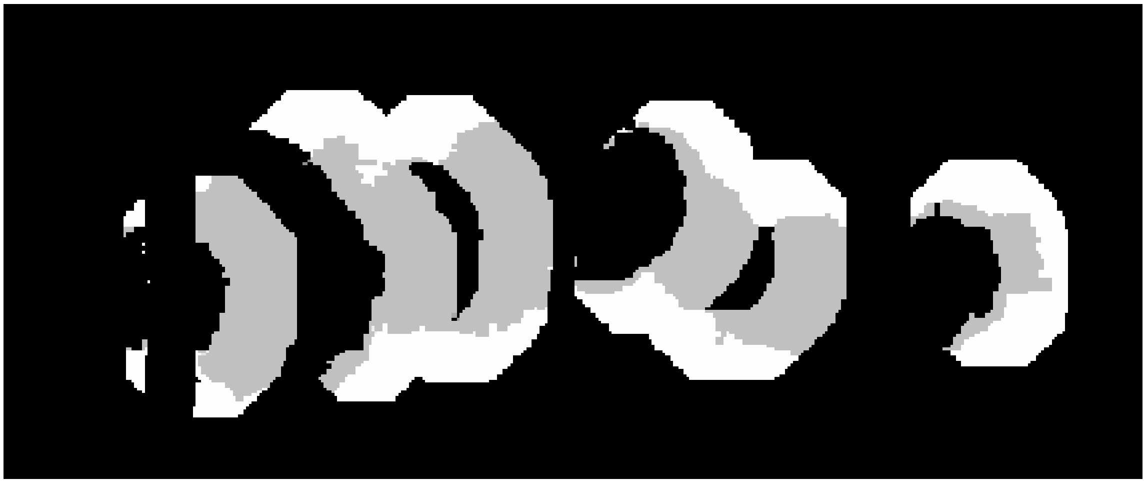


Figure 29. Growth frontiers at each time step

5.2.4. <u>Heterogeneities</u>

Other than the scribe and filament growth, various heterogeneities in the coating and sample material are visible in the CLSM images. The AA2024-T3 material used in this research has numerous composition elements including copper, magnesium, manganese, iron, zinc, and silicon, which are visible intermetallic particles in the images. Other heterogeneities such as surface holes, dust, and polishing particulate are also visible. Figure 30 provides a closer look at the CLSM image and illustrates the various heterogeneities that are present. The size of the area in the image is approximately 380μm x 480μm. Intermetallic particles generally appear as grey distinctive shapes, whereas surface holes are small black areas.

One of the primary goals of the micro scale approach is to determine if the heterogeneities can be used to predict the growth of filiform corrosion. There is evidence to suggest that the heterogeneities might influence filiform growth. The chemical makeup of the intermetallic particles and their role in the corrosion reaction makes them a likely candidate for growth features. Additionally, the holes in the surface of the material should cause the coating to adhere better in those locations. Because the filaments grow between the coating and the material, it would be a reasonable hypothesis that the filaments would be less likely to grow in areas where the coating bond is higher. Because of the possibilities of various relationships, it is important to be able to differentiate between heterogeneity types.

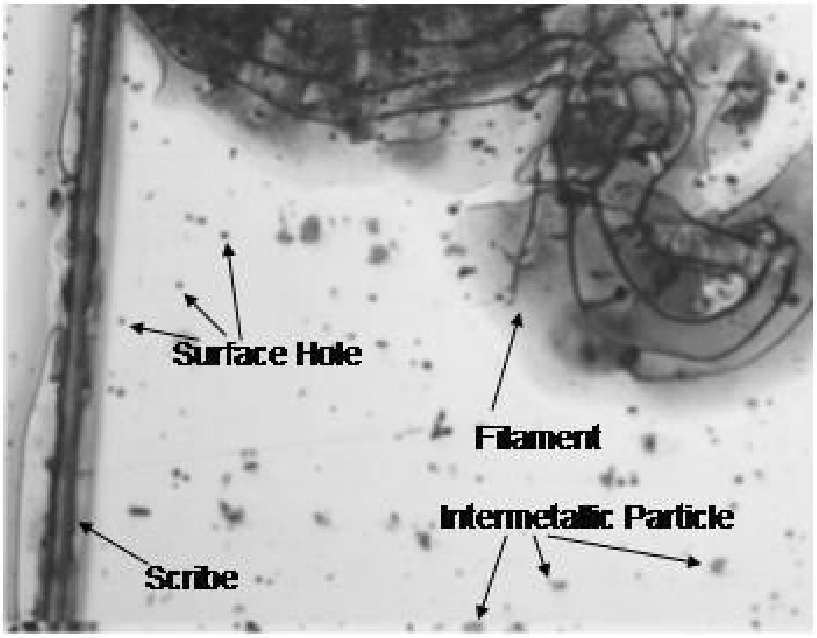


Figure 30. Sample heterogeneities visible in CLSM images

5.2.4.1. Heterogeneity Statistics

In order to use the heterogeneities as features, they must first be segmented from the image. The same segmentation methods used to segment the growth and scribe was used to segment the heterogeneities from the images for the nine areas in sample 1. The image at time $t = 0$ was used to segment the heterogeneities. Once the locations are known this information can be used at any time period. Segmentation produces a map of the placement of the distinguishable heterogeneities in each area. Once identified, various statistics were calculated in order to determine if separable groups exist. Multimodal distributions of size, for example, might suggest the existence of distinct classes of heterogeneity.

In addition to size, statistics for color and shape were calculated for the segmented objects. Color is determined by the intensity of the pixels in the object. A mean value is typically used due to the varying intensity value within a given object. The size of the object can be captured using the number of pixels as the object area, using the diameter of the circle with equivalent area, or by taking the minor or major axis length. Several metrics for shape were also considered. Eccentricity measures the ratio between the major axis length and the distance between the foci, where values of 1 indicate a straight line and higher values indicate a more circular object. Extent measures the ratio of the area of the heterogeneity to the area of the bounding box that contains it. Solidity measures the proportion of the pixels in the convex hull that are also included in the heterogeneity. The final statistic considered was the orientation of the heterogeneity, which is the angle between the x-axis of the image and the major axis of the heterogeneity.

Figure 31 displays the distribution for some of the statistics calculated for more than 110,000 heterogeneities segmented in the images. The pixel intensity is distributed similar to a Gaussian with mean intensity value near 180. The pixel area shows that most of the heterogeneities are small, with some outliers that are quite large. The distribution of extent values suggests that most of the objects are more regular in shape in that they typically cover more than 75% of the bounding box area. Although the statistics help to describe the heterogeneities present in the sample, it is not clear that these distributions imply an obvious grouping of heterogeneity type. However, size and color are assumed to be the most important attributes for distinguishing heterogeneity type given the ability to differentiate between intermetallic particles and surface holes based on intensity and area.

In the absence of obvious groupings, the intensity data was binned into nine different categories based on the pixel intensity, k. Heterogeneities were then classified as belonging to one of nine categories, L_j for $j = \{1,2,\ldots,9\}$. The definition of the heterogeneity categories is given below:

$$\begin{aligned}
&L_1 : k \le 100\\
&L_2 : 100 < k \le 120\\
&L_3 : 120 < k \le 140\\
&L_4 : 140 < k \le 160\\
&L_5 : 160 < k \le 180\\
&L_6 : 180 < k \le 200\\
&L_7 : 200 < k \le 220\\
&L_8 : 220 < k \le 240\\
&L_9 : k > 240
\end{aligned}$$

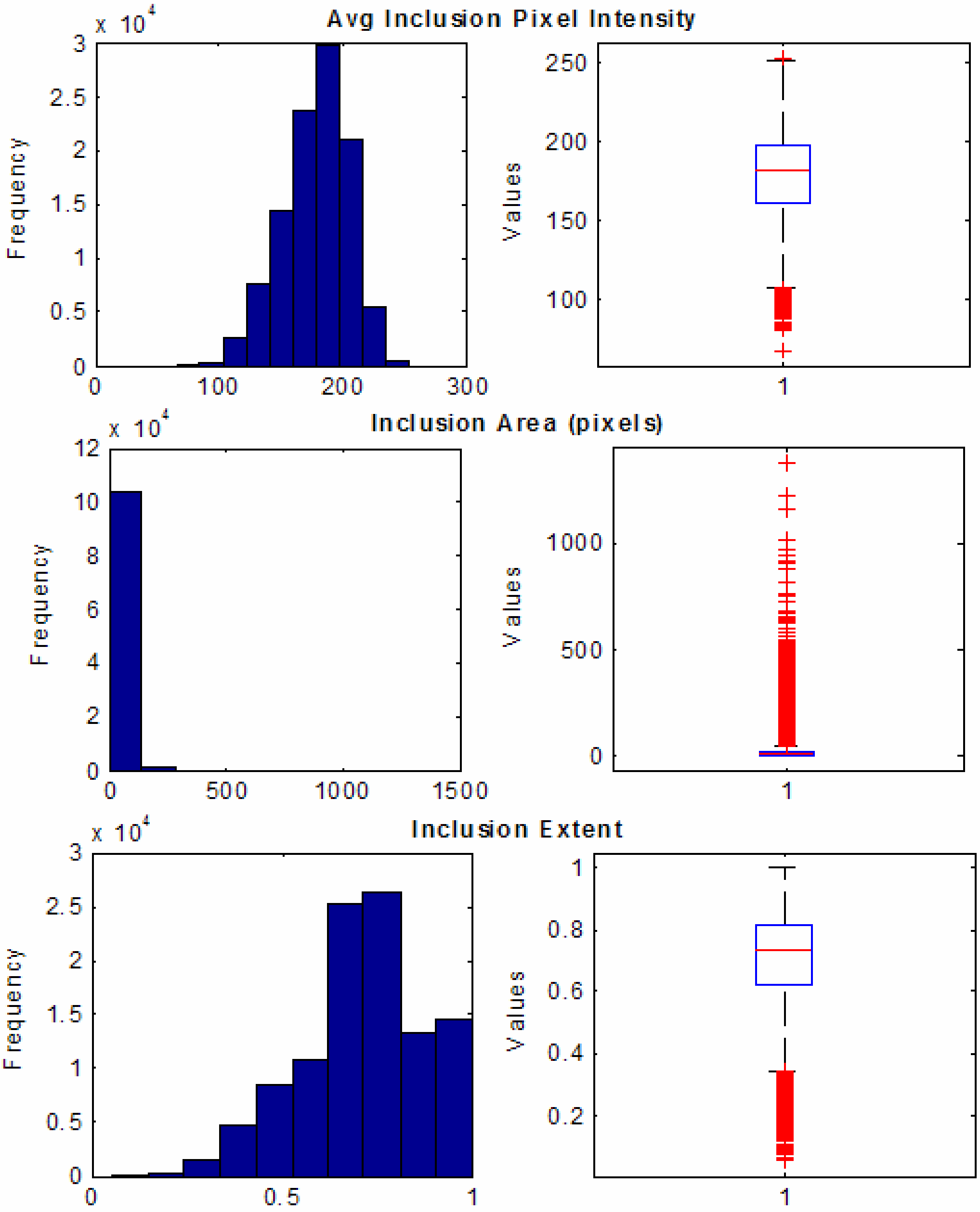


Figure 31. Statistics for heterogeneity color, size, and shape

5.2.4.2. Heterogeneity Densities

In order to use the heterogeneity categories as features, a feature value must be determined for each pixel location. The presence of a given category of heterogeneity could be used as a binary predictor; however, distance measures are more applicable especially given the previous hypotheses regarding the influence of intermetallic particles and surface holes on filiform growth. A typical distance measure would calculate the distance from a given pixel to the nearest heterogeneity location. This could be done for each pixel in the area and those values for each category of heterogeneity would represent the feature space. However, the distance measure makes no account for the size of the heterogeneity. The size values could be grouped in a manner similar to the intensity values, and additional distance values could be determined for each size class of heterogeneity, but this would greatly increase the number of features.

Kernel densities provide a manner for aggregating the distance and size information into a single feature value for each intensity category. Kernel density estimation involves placing a kernel at each event location. The kernel, K, can be any probability density function symmetric about the origin. The bandwidth, h, acts as a smoothing constant which determines the width of the kernel. Given a set of n observations, the density estimate for pixel location, x, is denoted $\hat{\mathrm{f}}(\mathrm{x})$. If we specify an event as each pixel belonging to a heterogeneity of the specific category, the resulting value of $\hat{\mathrm{f}}(\mathrm{x})$ accounts for distance to and size of heterogeneity.

$$\hat{f}(x) = \frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right) \tag{24}$$

For example, Figure 32 shows the L_2 heterogeneities for area 3 in sample 1. Placing a Gaussian kernel at each of the heterogeneity pixels produces the density estimate shown in the image on the right. The filiform growth for the area is outlined in black as a reference. Similar densities were calculated for each category of heterogeneity and each area of sample 1. The values for the densities can then be used as the feature values at each location.

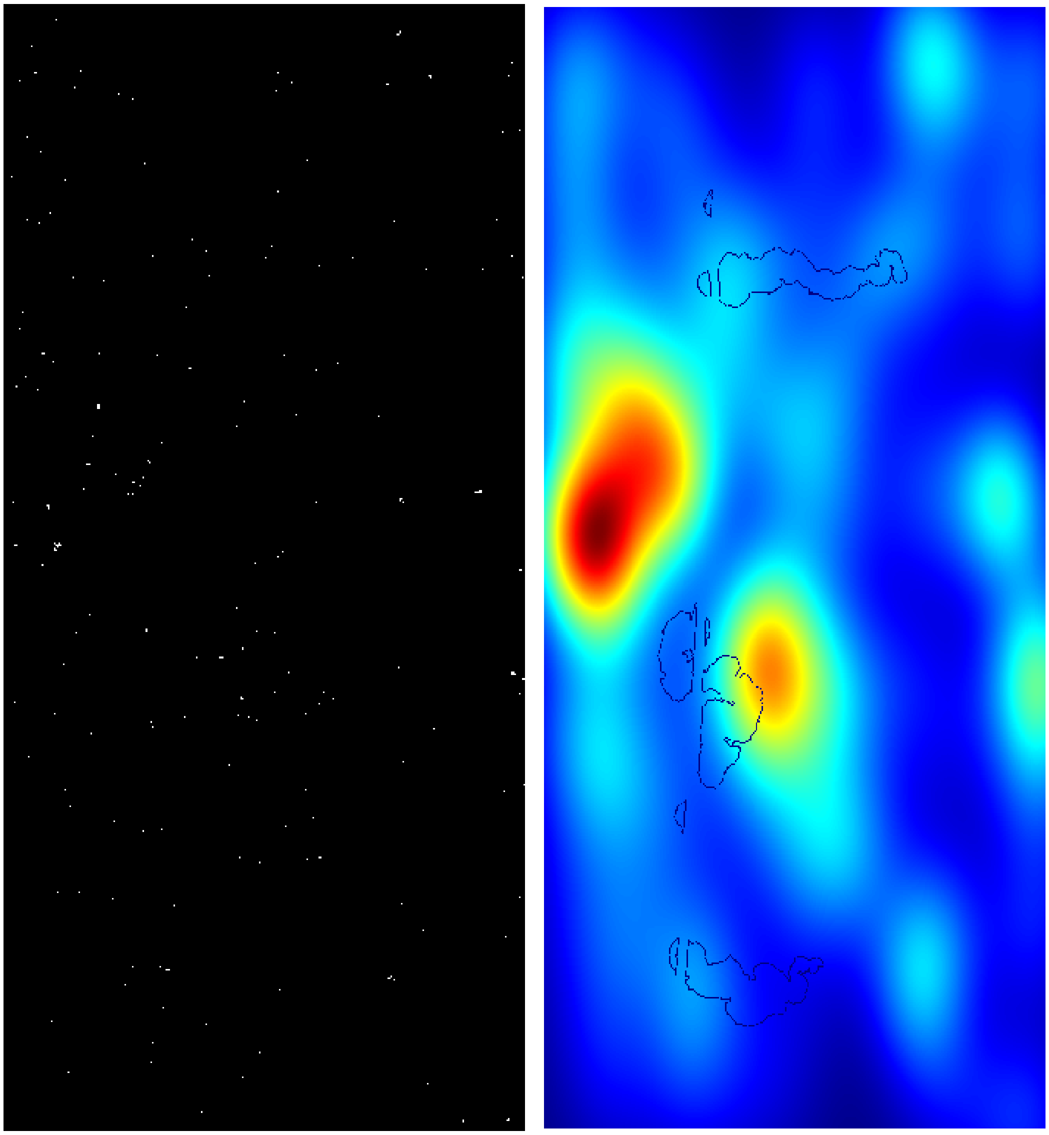


Figure 32. L_2 heterogeneities in area 3 and resulting kernel density estimate

A great deal of research has been done on the selection of kernel functions and bandwidth values. The results of these studies show that the choice of kernel function is of significantly less importance than the value of the bandwidth used (Scott 1992). Many methods have been suggested for determining the optimal bandwidth based on different assumption of the underlying density $f(x)$ and different optimality criteria. A common criterion is the asymptotic mean integrated square error (AMISE). For a Gaussian kernel and AMISE criterion, the optimal bandwidth h_d^*, for the d^{th} dimension of x can be determined using the d^{th} dimension standard deviation of x, σ_d, where $d = \{1,2,\ldots,p\}$ (Scott, 1992). For this approach the densities are 2 dimensional over the material surface. The density in Figure 32 uses a bivariate Gaussian kernel with AMISE optimal bandwidth $h_1^* = 230$ and $h_2^* = 141$.

$$h_d^* = \left(\frac{4}{p+2}\right)^{1/(p+4)} \sigma_d n^{-1/(p+4)} \tag{25}$$

CHAPTER 6. MACRO SCALE RESULTS

The results for the macro scale data is presented in this chapter. Results are first given for the fitting of model parameters using the training data. The test data is then used to evaluate the selected models against a model of random growth.

6.1. Training and Test Data

As described in Chapter 4, the images for growth from samples 1 and 2 were used as training data for the macro approach, and data from sample 3 was kept as a test set. All 19 of the derived features were considered for use in each model type. The training set consisted of over 280,000 observations and the test set contained nearly 150,000 observations for filiform growth from day 1 to day 2. The resolution of the images made it impossible to segment individual filaments, therefore and interacting particle model for a single filament was not attempted. Due to the lack of features in the initial image, other than distance to the scribe, a feature-based analysis of initiation was also not attempted. The primary goal of the work at this scale was to investigate the ability of the derived features to explain the filiform growth.

6.2. Model Fit

The various model parameters were fit using the training data. Models were selected using the metrics for precision, recall and f-measure. Logistic regression, classification trees, and generalized additive models were implemented using the Insightful Miner Version 3.0 which runs with SPlus Version 6.2. Kernel density classification was implemented using Matlab Version 7.0.

6.2.1. Logistic Regression

All of the features described in the previous section were included in the model as well as all 2-way and selected 3-way interaction terms. The coefficients for the predictor variables are estimated using maximum likelihood. The output for the logistic regression model is provided in Appendix C. A Wald statistic was calculated for each of the predictor variables to determine the significance of including each term. Table 4 shows the 20 most significant variables in terms of their Wald statistic. The values of the Wald statistic for these variables are all significant at the 99% confidence level (i.e. the p-values are all less than 0.01). The Wald statistic is the square of the t-statistic and generally provides a better measure for how the variable contributes to the model as a whole. Although problems of bias with the Wald statistic have been published (Menard, 2001), the coefficients for this application are small and the availability of the statistic make its use acceptable. The Wald statistic suggested very little reduction in predictor variables. Some of the most significant variables include the distance from initiation point, distance to nearest corrosion point, angle to nearest corrosion point, the number of pixels within a 17x17 region, and the angle between the nearest corrosion point and the initiation point.

The logistic regression produces a probability of corrosion value for each pixel observation. A threshold value must be selected in order to classify the pixel as corroded or not. Low threshold values result in a higher false positive rate and high thresholds produce high false negative rates. Because we are primarily concerned with identifying corrosion, recall and precision are used as metrics. Recall represents the percentage of truly corroded pixels that are accurately classified. Precision represents the percentage of predicted corroded pixels that are accurately classified. Figure 33 shows how the recall and precision values change as the threshold is increased. This chart suggests an optimal threshold of approximately 40% to maximize recall and precision simultaneously. Therefore, the training data suggests that the logistic regression model will result in approximately 68% for recall and precision.

Variable Name	Wald Statistic
init_d	1,764.79
pc_ang:ci_ang:ci_d:gap	605.89
ci_d	356.94
ci_ang:ci_d:gap	313.33
pc_ang:ci_d:gap	284.5
pc_ang:ci_d	239.44
numcorr_17	230.87
pc_ang:ci_ang:ci_d	205.63
ci_d:dang_ratio	196.91
corr_d	184.44
sameside	183.79
pc_ang	172.81
pc_ang:ci_ang:dang_ratio	159.75
sizecorr	157.6
pc_ang:ci_ang:gap	151.77
pc_ang:ci_ang	141.29
anginit	120.02
ci_ang:dang_ratio	108.25
pc_ang:dang_ratio	88.53
corr_d:sizecorr	87.11

Table 4. Wald statistics for macro scale logistic regression model

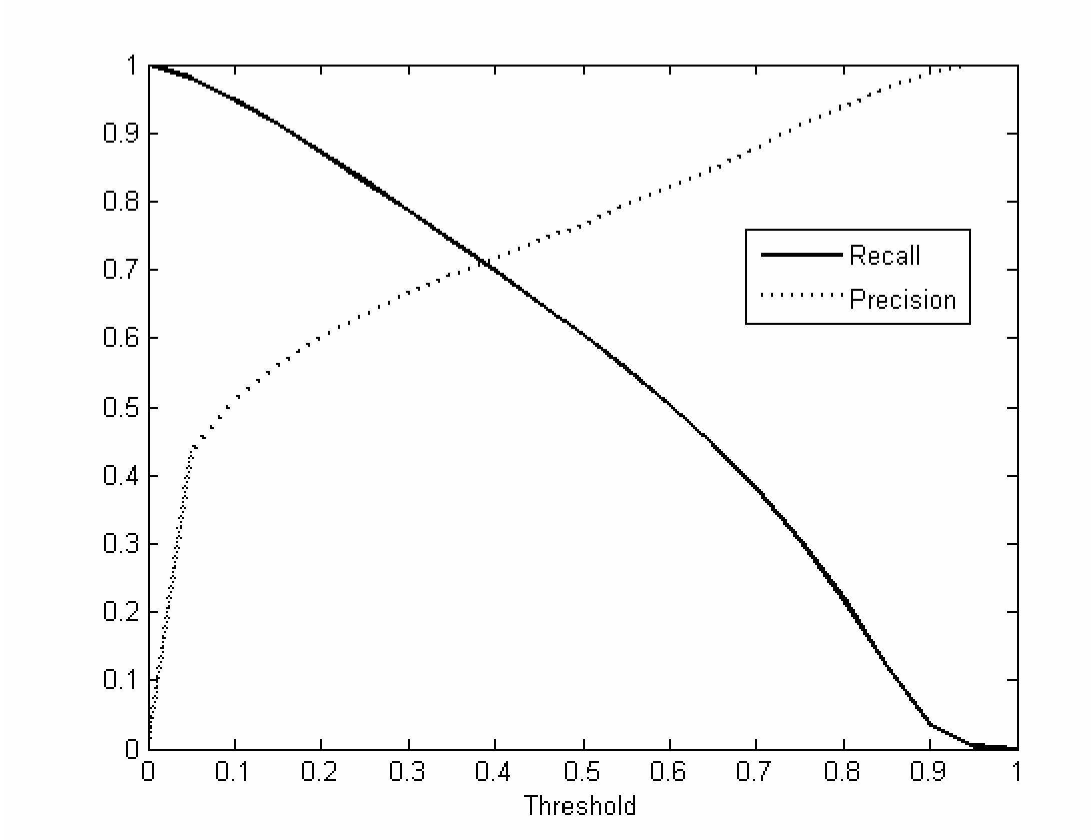


Figure 33. Precision and recall plot for logistic regression model

6.2.2. Classification Tree

For this study, ensemble trees were constructed using the training data. Both the Gini index and entropy metrics were considered, but the Gini index provided better accuracy results for the training data, therefore, the entropy metric was not used further. The amount of impurity that is removed by splitting on a given feature provides a metric for the significance of that feature in predicting the response classification. Figure 34 shows the relative importance of the features used in the ensemble trees. The derived feature dang_ratio, as well as the distance to nearest corrosion (corr_d) and distance to nearest initiation point (init_d) provided the largest reduction in Gini index score. These results are fairly consistent with the logistic regression coefficients, although the order of significance is changed. The inclusion of interaction terms in the logistic regression output is a primary source of difference in the result. The interactions are accounted for in the construction of the classification tree. The classification tree achieved 68% and 69% in recall and precision for the training data.

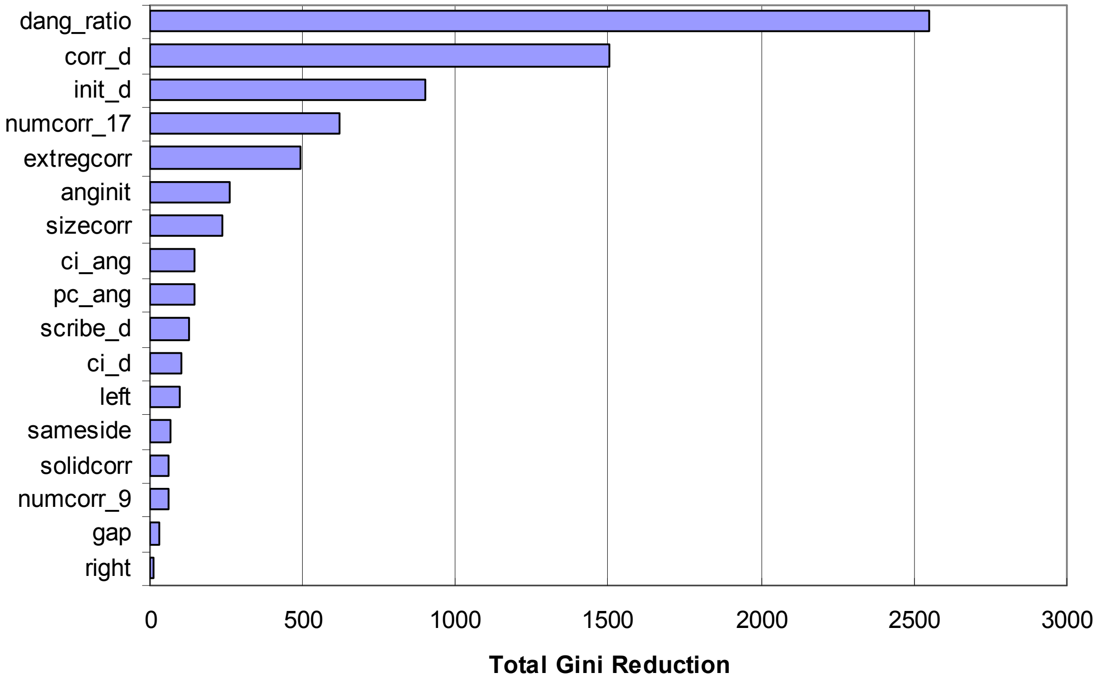


Figure 34. Feature importance from classification tree

6.2.3. Other Models

Generalized additive models (GAM) and kernel density classifiers were also considered for use in this application. The GAM requires a smooth function to be fit to each predictor variable, which when included in the additive model captures the nonlinearities present in the data. A cubic spline with 3 degrees of freedom was used as the functional form for each predictor. Figure 35 shows the spline approximations for selected features. The results with the training data once again showed that all of the predictor variables were significant at the α=0.01 level. The results for GAM in terms of recall and precision were 70% and 67%.

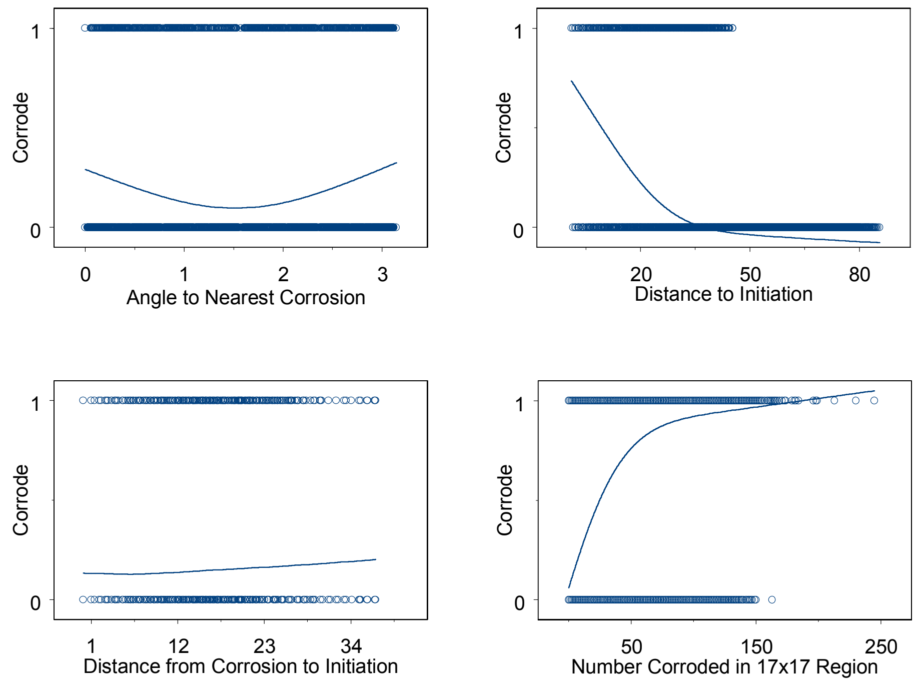


Figure 35. Spline approximations for selected predictor variables

Classification with kernel density estimates (KDE) required some transformation of the original feature set. The major drawback to classification with KDE is it's reliance on the entire training data set. The KDE model is the data set, and fitting of the model is done at evaluation (Hastie *et al.*, 2001). Because of computational and visualization considerations, we chose to reduce the dimension of the data set. Principal component analysis (PCA) provides an accepted method for shrinking the number of features and can simplify KDE (Scott, 1992; Cooley and MacEachern, 1998). In addition, it ensures that regardless of any correlation in the original feature variables, the new principal component features will be uncorrelated. The first 3 principal components were considered for this data. Cumulatively, they explain 63% of the variability in the data. The kernel density estimates were fit to the 3 principal components using Gaussian kernels. The bandwidth for the kernels is the AMISE optimal bandwidth described in Chapter 5.

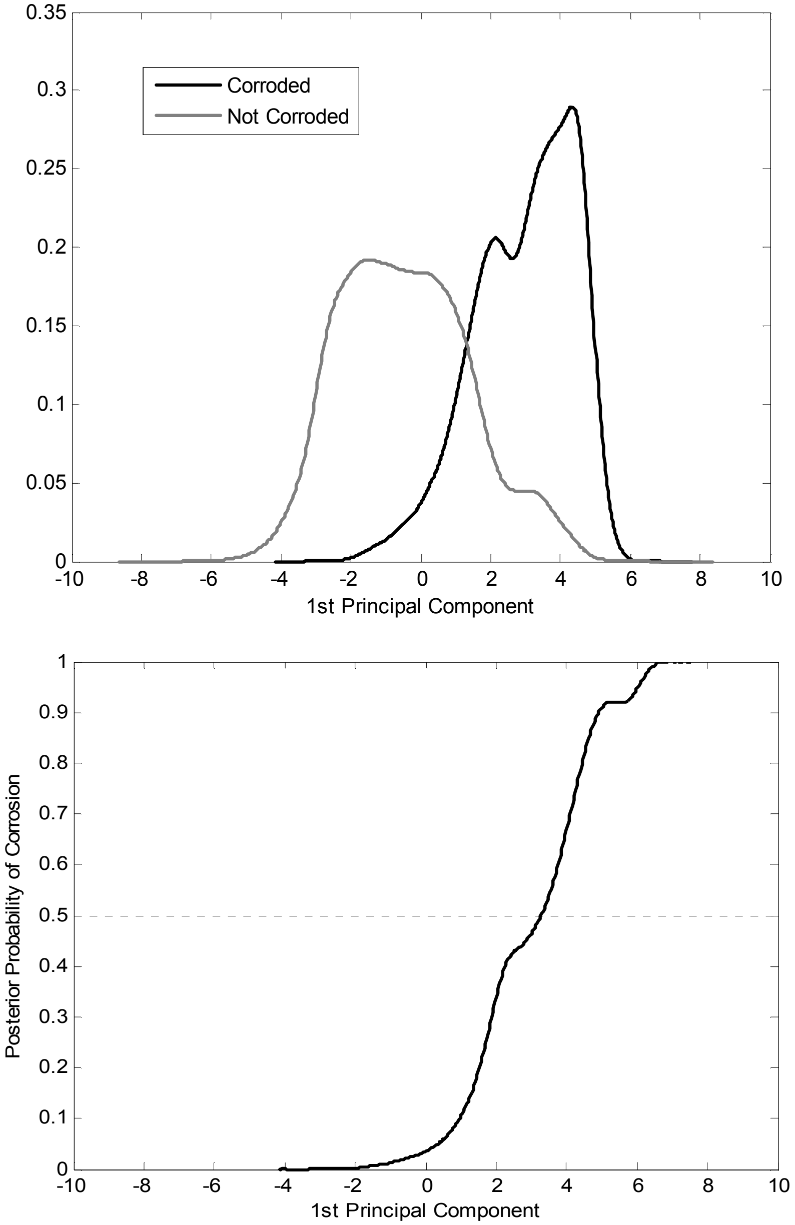


Figure 36. One-dimensional kernel density classification with principal components

KDE was used to estimate the population class density for pixels being corroded and not corroded. Figure 36 shows the class densities in one dimension, using only the first principal component. However, to determine the posterior probability of a pixel corroding, we need to adjust for the prior probability of corrosion. In this case the sample proportion of corroded pixels was about 15%. The posterior probabilities can be calculated using Bayes' theorem. The one-dimensional posterior probability shown in Figure 36 suggests that observations with 1st principal component values > 3 should be classified as corroded.

We calculated the posterior probabilities from the 3-dimensional kernel density estimates. Pixels in the training data were classified as corroded when the posterior probability > 0.5. These results were then compared to the true class labels. The results for KDE were significantly lower than the other methods considered. The recall and precision values were 43% and 76%.

6.3. Model Evaluation

The results for the training data are summarized in Table 5. Based on the performance with the training data set, the logistic regression, classification tree, and GAM were considered for model choices. All of these models had similar recall and precision results and were relatively efficient given the size of the data. Each fitted model was used to predict the growth of corrosion on the test data sample. Because of the low incidence of corrosion, we are primarily concerned with accurate prediction of corroded pixels.

Model	Precision	Recall	F-Measure
Logistic Regression	65.4%	69.6%	67.4%
Classification Tree	68.7%	67.8%	68.2%
Generalized Additive Model	66.7%	70.3%	68.5%
Kernel Density Classification	76.1%	43.2%	55.1%

Table 5. Training data classification results

6.3.1. Random Model

Because of the lack of a suitable baseline model, an informed random model was developed as an additional measure of comparison. The random model uses the corrosion rate calculated from the training data to determine the number of pixels that will corrode in the test data. The rate of corrosion for the training data was 6.23% which results in 31,127 predicted corroded pixels for the test data. The random model defines eligible pixels as those that neighbor a currently corroded pixel. The model then randomly selects a single eligible pixel to corrode. Then the set of eligible pixels is updated and the process continues until the desired number of corroded pixels is reached. The result is predicted growth that is uniformly mounded around the current corrosion.

Multiple iterations of the random model were run to create a credible interval of random corrosion. A credible interval for corrosion growth can be constructed using the number of times a given pixel is randomly selected to corrode divided by the total number of iterations. Figure 37 shows the results of 20 iterations of the random model on the left end of the test sample. Greater intensity represents pixels that are corroded in more iterations (i.e. more likely to corrode according to the random model).

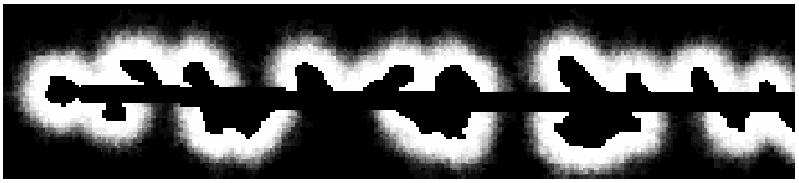


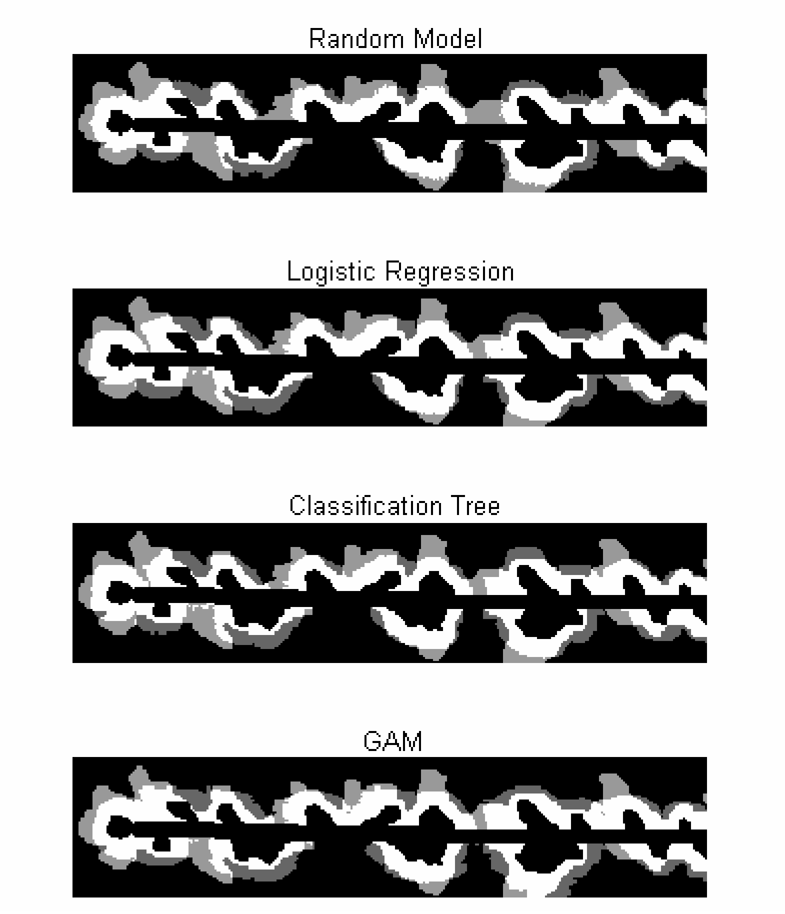
Figure 37. Credible interval for random model growth

6.3.2. Predictive Results

The fitted models were evaluated based on their ability to predict corrosion growth on the sample 3 test data. Figure 38 shows a portion of the sample with a comparison of predicted and true corrosion growth. The white pixels are correctly classified corrosion. The light grey pixels were incorrectly classified as non-corroded (type I error), and the dark grey pixels are incorrectly classified as corrosion (type II error). Table 6 provides the precision, recall and F-measure classification results for the test data.

As expected, the random model predicts uniform growth around the current corrosion, but is unable to capture areas where the filament "fingers" in a particular direction. The other models, however, also failed to capture areas of directed growth. The logistic regression, tree, and GAM models appear to capture separation between corrosion objects better than the random model which tends to force the objects to merge together. However, the predicted image shows that the model is not a very accurate representation of the true corrosion process.

The recall and precision results quantify the results shown in the predictive images. The random model captures the least amount of true corrosion, but has the fewest type II errors. The GAM model has the highest recall, but results in the most type II errors. The GAM model had fairly high recall and precision values and in terms of the F-measure, was the best predictor. Despite the relatively high values for recall and precision, none of the proposed classification models adequately captured the growth process of filiform corrosion. Although the features used in the models were statistically significant, they are insufficient to describe the underlying process.



Random Model

Logistic Regression

Classification Tree

GAM

Figure 38. Predicted model results vs. actual growth
(White: correctly classified, Light Grey: type I error, Dark Grey: type II error)

Model	Precision	Recall	F-Measure
Random Growth	75.7%	60.6%	66.8%
Logistic Regression	72.9%	63.8%	68.0%
Classification Tree	71.8%	69.9%	70.7%
Generalized Additive Model	70.7%	71.7%	71.2%

Table 6. Test data classification results

CHAPTER 7. MICRO SCALE RESULTS

Results for the initiation and growth models using the micro scale data are given in this chapter. The initiation results are presented first with the details of the initiation model components. The growth model components are estimated and the results provided for the micro scale data from sample 1. A validation of the growth model is also presented using new data from a second sample.

7.1. Initiation

The goal of this research is to develop a model for filiform corrosion evolution, which includes the initiation and growth of filaments. The model for initiation, presented in Chapter 3, is based on a previously developed feature-based model for point process prediction. The results presented in this section illustrate the applicability of the feature-based methodology to the filament initiation process. The analysis of the initiation results is focused primarily on gaining insight into the initiation process and determining the influence of the heterogeneity features on the location of filament initiation.

7.1.1. Data

Initiation sites were identified on each of the 9 areas of sample 1. A total of 40 distinct locations were determined using the images taken after 6 hours of exposure. Given the coarseness of the time scale for imaging, the exact location of initiation is unknown and cannot be determined using the available data. However, the centroid location of the visible growth at time $t = 6$ provides a reasonable estimate of the actual location. The images in Figure 39 show the CLSM image of a filament from sample 1 at time $t = 6$. The second image illustrates the segmented growth of the initiated filament with the scribe pixels removed so that the centroid location can be seen (indicated by a +).

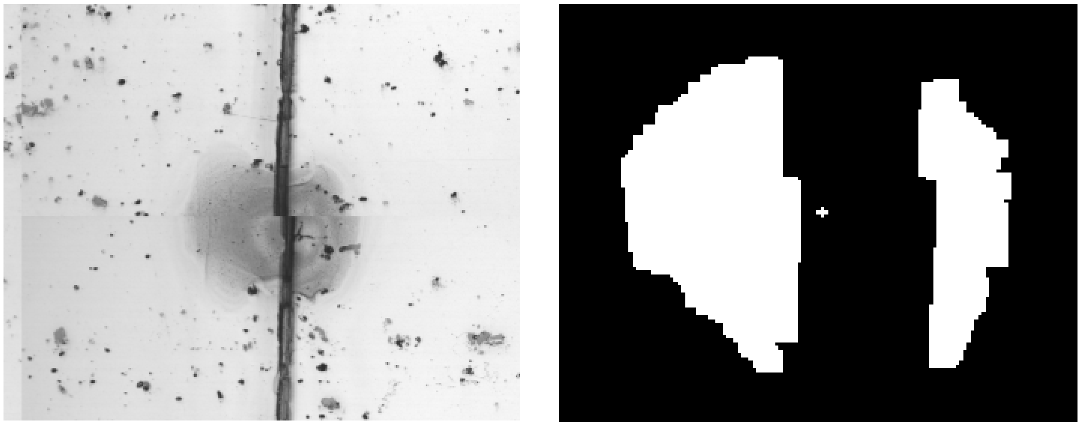


Figure 39. Filament initiation and determination of location using centroid of growth

The centroid location of each filament serves as the initiation site and is marked by the values of the 9 heterogeneity features described in Chapter 5. As with the growth model, the relationship between the heterogeneities in the sample and the initiation of filaments was unknown. Other factors, specifically those associated with the coating and scribe mechanism, may influence the number and location of initiation sites, however, data for those features was not available for this work.

The initiation model consists of a spatial and temporal transition density. Initial analysis of the images from sample 1 suggested that all initiation occurred in the first image at time $t = 6$, because no other initiation was visible in subsequent images. This hypothesis was tested and confirmed using additional samples with the goal of capturing initiation at an earlier state so the true location could be better identified. Connor Parker, MS student in the department of Material Science and Engineering at the University of Virginia, performed the experiments, which consisted of two additional samples of AA2024-T3. The samples were polished to a 1μm finish, coated with epoxy polyamide, and then scribed using a scalpel creating a 30mm scratch on each surface. The panels were then exposed to concentrated HCl for 30 seconds and placed in a temperature and humidity chamber at 40°C and 80% relative humidity. The first sample was removed and inspected under a microscope for filiform initiation after 1, 2, 5, and 7 hours of exposure in the chamber. The second sample was inspected after 2, 4, and 6 hours of exposure. The first sample had visible initiation at 7 hours and the second sample had visible initiation at 6 hours. Individual initiation sights were imaged on both samples using the CLSM. Given that all initiation occurred at approximately the same point in time, the primary component of interest is to determine the location of initiation sites with the spatial transition density.

7.1.2. Spatial Transition

The spatial transition density described in Chapter 3 is estimated by the 1st order spatial transition density, $\psi_n^{(11)}(x_{n+1} \mid \chi_n)$, which determines the likelihood that the next initiation will occur at a given location in feature space as a function of the feature values of previous events. The spatial interaction between initiation events is not addressed due to the fact that the data comes from different areas along the sample and the limited number of initiation locations in each area. Moreover, we are primarily interested in determining the role of the heterogeneity features in the process which is captured in the 1st order transition density, and the assumption of no spatial interaction is reasonable given the observed initiation locations.

7.1.3. Predictive Results

The spatial transition density is estimated using kernel density estimates from events in feature space. Because the data set consists of only 40 events over 9 areas, the dimension of the feature space was reduced to allow better coverage of the space. For each area, a KDE was calculated in a 2-dimensional feature space using events from all other areas. Each possible combination of feature pairs was considered in the KDEs. The KDEs were constructed using Gaussian kernels and the AMISE-optimal bandwidth rule described in Chapter 5. An example of the feature-based KDE is provided in Figure 40. The figure shows the estimated transition density for area 9 using features 8 and 9 (densities of light-colored heterogeneities). The initiation points are highlighted by the arrows, and appear to be located in regions of higher predicted probability. The locations of the initiation points in a given area are compared to the KDE for that area using percentile score and likelihood metrics to determine which features, if any, provide increased predictive ability and improvement upon a prediction from a uniform geographic transition density.

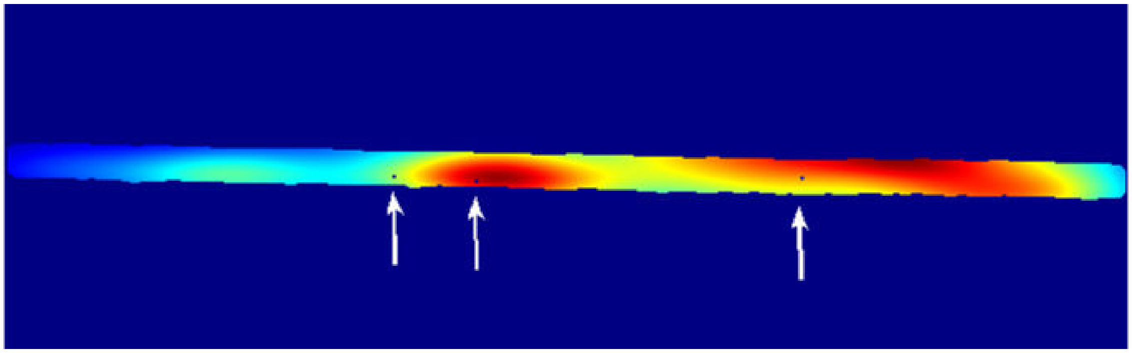


Figure 40. Initiation transition density for area 9 using features 8 and 9

7.1.3.1. Percentile Score

Percentile scores were calculated according to the definition provided in Chapter 4, for each initiation point in each area, and for each pair of features. Given the number of initiation locations in area a, n_a, and the percentile score of initiation location i in area a, $\pi_a(i)$, we calculate the mean percentile score across all areas in sample 1, $\bar{\pi}$.

$$\bar{\pi} = \frac{\sum_{a=1}^{9} \frac{\sum_{i=1}^{n_a} \pi_a(i)}{n_a}}{9} \tag{26}$$

The mean percentile score across all areas is provided in Figure 41 for each pair of features. The highest values are clustered around the feature pairs that include feature 3, specifically (1:3), (2:3), (3:4), (3:5), (3:6), (3:7), and (3:8). This would suggest that the heterogeneities with intensity levels corresponding to feature 3 are the most important to initiation. The chart also suggests that the heterogeneities corresponding to feature 9 are the least important. A drop in mean percentile score occurs in each of the feature pairs containing feature 9. For comparison, the mean percentile scores for each individual area are provided in charts in Appendix D, which show fairly consistent results across the areas.

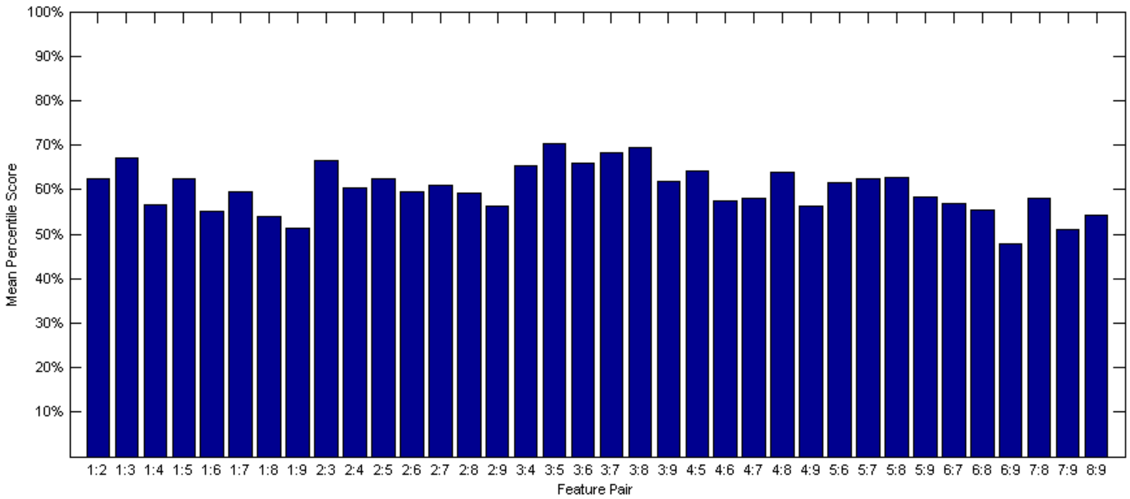


Figure 41. Mean percentile score across all 9 areas

7.1.3.2. Likelihood

The likelihood of each initiation location was determined from the estimated spatial transition density, and the product of the likelihood for each initiation point is the likelihood for the observations in area a, $L_a(\theta)$. We use the likelihood values in a minimum Bayes risk method for the 2-class classification decision to determine whether the observations are more likely generated by the estimated feature-based spatial transition density or from a uniform spatial density (implying a random distribution of initiation locations). Assuming equal prior probabilities and misclassification costs, the decision rule is determined by the likelihood ratio $\Lambda(y)$. If $\Lambda(y)$>1, then we conclude that the observations are generated from the feature-based density, $\hat{f}(y;\theta_X)$, otherwise, we conclude that the observations are more likely generated from a uniform spatial density, $\hat{f}(y;\theta_V)$.

$$\Lambda(y) = \frac{\hat{f}(y;\theta_X)}{\hat{f}(y;\theta_V)} \tag{27}$$

For each area, we use the KDE built from the features of the initiations in other areas, $\hat{f}(y;\theta_X)$, and evaluate the likelihood of the initiation locations. Again, each pair of features is used to construct a KDE of the spatial transition density. Figure 42 shows the result of the likelihood ratio decisions over all areas for each feature pair. As suggested by the percentile score results, the feature pairs that include feature 3 offer improved predictive ability. In all but 1 area (area 3 for features 3:5, and area 6 for features 3:6), the observed initiation locations are more likely generated by the feature based density than the uniform density. The poor performance of feature 9 is also consistent with the percentile score results; there are very few areas where the feature-based density with feature 9 is likely to produce the observed initiation locations.

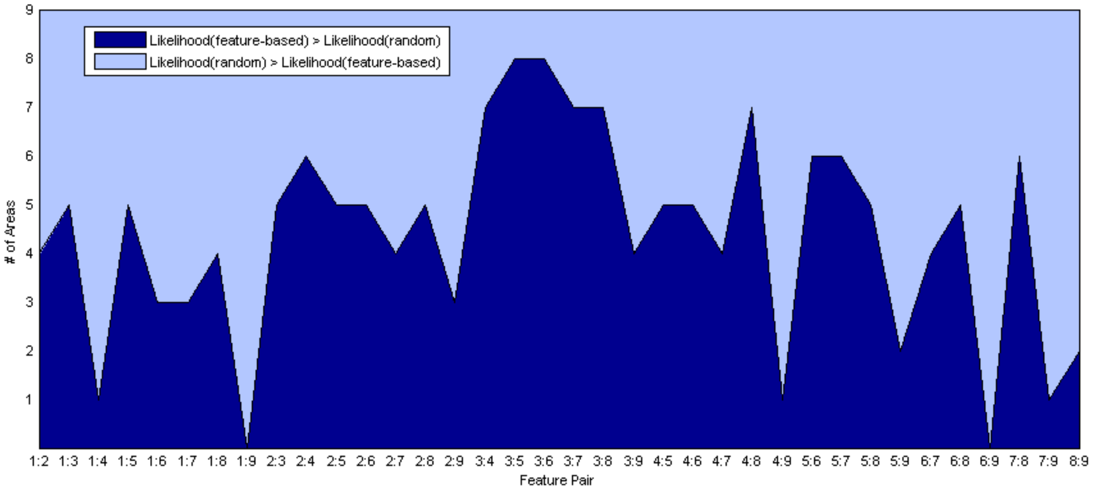


Figure 42. Initiation likelihood ratio decision over all areas

Under the assumption that the feature-based transition density, $\hat{f}(y;\theta_X)$, is the true transition density for the process, we can generate a distribution of expected likelihood values using simulation to produce new initiations. The likelihood of the observed initiation locations can then be used as a test statistic in a hypothesis test of the true underlying transition density.

In each area, and for each pair of features, we performed two hypothesis tests using the distribution of likelihood values obtained from simulation. In the first test, we assume that the observations are generated from the feature-based density, $H_o : f(y;\theta) = \hat{f}(y;\theta_X)$ vs. $H_a : f(y;\theta) \neq \hat{f}(y;\theta_X)$. Sets of initiation locations are generated from $\hat{f}(y;\theta_X)$, and the likelihood values are calculated and kept to create a distribution of expected likelihood values if $f(y;\theta) = \hat{f}(y;\theta_X)$. A total of 10,000 likelihood values were generated in the simulation. The likelihood of the actual observed locations is then compared to the simulated distribution. Because this is a two-sided test, we fail to reject H_o at a 5% significance level if the observed value falls within the 2.5-97.5% range of the distribution values. The results of this hypothesis test are presented in Figure 43. The chart shows the percentage of areas where we reject H_o for each feature pair. Once again, there are very few areas where we reject H_o when feature 3 is included in the feature pair. When the density incorporates feature 9, however, we reject H_o in almost every area.

For comparison, we performed a similar hypothesis test, assuming that a uniform spatial density is the true underlying density, $H_o : f(y;\theta) = \hat{f}(y;\theta_V)$ vs. $H_a : f(y;\theta) \neq \hat{f}(y;\theta_V)$. Again, 10,000 simulated sets of initiation locations were generated from a uniform spatial density to create a distribution of expected likelihood values if the process were spatially random. The results of this test are provided in Figure 44, which shows the percentage of areas where we reject H_o. In analyzing the results from the two tests, we seek to find low values for feature pairs in Figure 43 with corresponding high values in Figure 44. The results show very few areas and feature pairs where we can reject the hypothesis that the observations are generated from a uniform transition density. However, the feature pairs (2:3), (3:5), and (3:8) provide the best alternatives for the feature-based transition density. In the case of feature pair (3:5), there is only one area (area 3) where we reject the hypothesis of the feature-based density as opposed to 4 out of 9 areas where we reject the uniform spatial density.

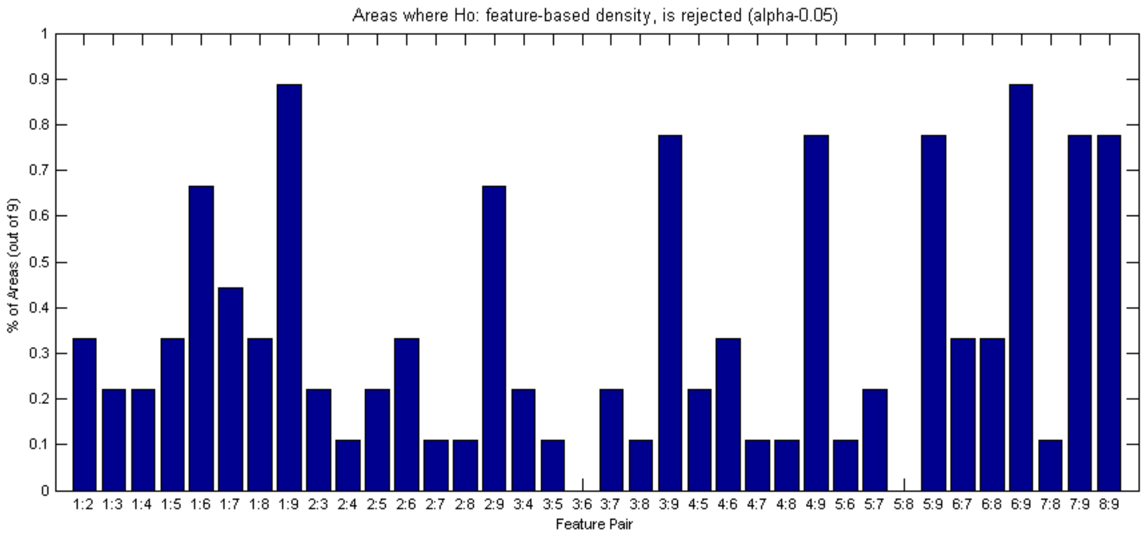


Figure 43. Hypothesis test for feature-based transition density

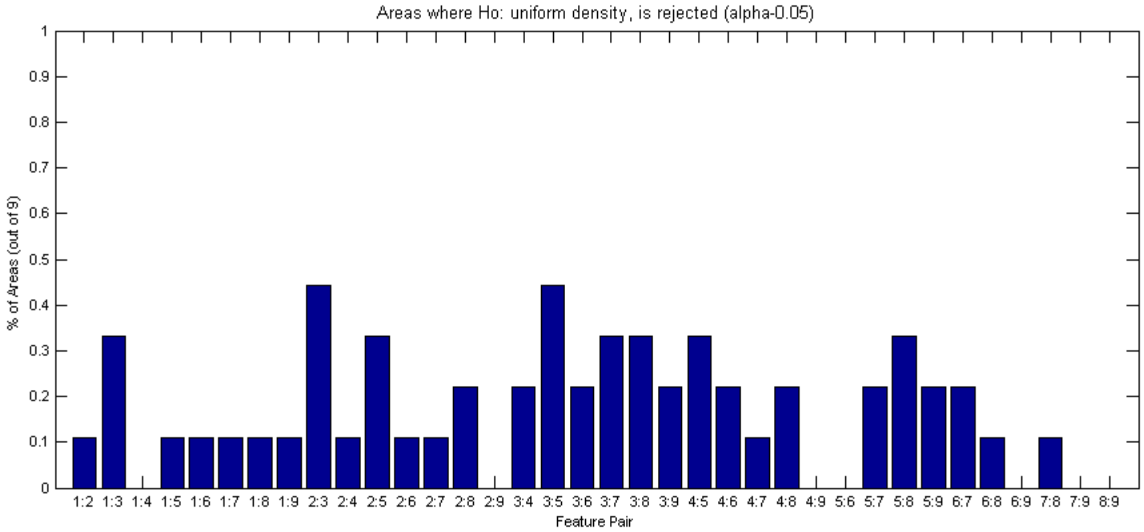


Figure 44. Hypothesis test for uniform transition density

7.2. Growth

In this research, more focus has been given to the prediction of filament growth in order to demonstrate the extension of the previously proven point process model used for initiation, into a feature-based model for growth prediction. The micro-scale approach was designed around the desire to determine the relationship between the heterogeneities visible in the samples and the growth of filiform corrosion. The heterogeneities are incorporated into the feature-based growth model for prediction. The results for the prediction of growth are generated using the interacting particle model described in Chapter 3. The primary components of the model are the growth probability function and the local transition function, which together control the predicted transition of cell states.

7.2.1. Growth Probability Function

The growth probability function is used to determine the likelihood that a given pixel corrodes as a function of the feature values at the pixel location. The functional form for this application is unknown, therefore, a variety of classification models are considered. All of the models are fit using the feature data described in Chapter 5, with the Insightful Miner version 3.0 software application.

Heterogeneities were categorized according to their average pixel intensity and named {L1, L2, …, L9}, where L1 represents the darkest and L9 represents the lightest heterogeneity group. Data was collected at each time interval using only those regions adjacent to current growth. Therefore, only those pixels that were actually candidates for corrosion at the next time step were included in the data set. The training data set consisted of the images from areas {3, 4, 6, 8, 10, 11} and produced 1,435,007 observations. Areas {5,7,9} were used for the test data set, and included 683,733 observations.

7.2.1.1. Logistic Regression Results

Logistic regression models were constructed using all main effects features as well as all two-way interaction terms. The logit link function was used to transform the response variable and the parameters are maximum likelihood estimates produced by the iteratively re-weighted least squares algorithm. Coefficients are based on normalized feature data, centered on the mean and scaled by the standard deviation of the feature. The results in Table 7 show the coefficient estimate and corresponding Wald statistic for the 20 most significant variables in the 1st order logistic regression model fit to the training data. The p-values for the variables in the table are all less than 0.01. The most significant main effects are {L1, L2, L3, L6, L8} and many of the interaction terms with these and other features are also significant. The coefficients suggest that as the intensity of L1, L2, and L8 heterogeneities increase, the probability of corrosion decreases. Similarly, the predicted probability of corrosion increases as the intensity of L3 and L6 heterogeneities increases.

The results for the 1st order logistic regression model, as well as the other models are provided in Table 9. The model resulted in an overall accuracy of only 57% for the training data set. The model had a very high recall rate indicating that almost all of the truly corroded cells were correctly classified; however, the low precision value indicates a high false positive rate.

Variable Name	Coefficient Estimate	Wald Statistic
L5:L6	-0.22	7,426.13
L2:L6	-0.21	7,323.40
L7:L8	0.22	4,424.21
L3:L6	0.18	4,327.37
L3	0.19	4,070.91
L6	0.14	3,079.17
L1:L6	0.2	2,920.33
L4:L5	-0.15	2,628.18
L2:L7	-0.2	2,447.57
L5:L7	0.19	2,368.54
L8	-0.28	2,325.98
L2:L4	-0.16	2,264.57
L1:L4	0.14	2,181.51
L1:L8	-0.34	1,760.98
L3:L5	0.11	1,714.43
L1	-0.12	1,133.26
L6:L7	-0.11	1,097.30
L1:L7	0.25	1,014.46
L2	-0.09	971.34
L2:L8	0.11	789.76

Table 7. Significant features from micro scale 1st order logistic regression

A 2nd order logistic regression model was also considered in order to capture nonlinearities in the feature variables. The model included all main effects, squared main effects and two-way interaction terms. The coefficients for the most significant variables are provided in Table 8 and complete results are given in Appendix F. The p-values for the variables listed in the table are all less than 0.01. The most significant main effects are {L1, $L1^2$, $L2^2$, L3, $L3^2$, L6}, with $L1^2$, L3, $L3^2$, and L6 having a positive correlation to the likelihood of corrosion and L1 and $L2^2$ having a negative correlation. The resulting relationship with L1 generally causes the probability of corrosion to decrease as the intensity of L1 increases. The estimates for L3 cause a more nonlinear relationship, with the probability of corrosion increasing dramatically as the intensity of L3 heterogeneities increases. The accuracy of the 2nd order model for the training data (Table 9) is improved over the 1st order model. Although the false positive rate is better, the recall rate is lower, resulting in a slightly decreased F-measure value.

Variable Name	Coefficient Estimate	Wald Statistic
$L1^2$	0.06	6,952.33
L1	-0.40	5,802.77
L6	0.19	5,262.03
L1:L8	-0.62	4,539.27
L2:L7	-0.33	4,339.91
L5:L7	0.27	4,201.13
$L2^2$	-0.16	4,149.39
$L3^2$	0.13	3,131.25
L1:L2	0.17	2,901.88
L3:L5	0.19	2,703.04
L1:L4	0.18	2,615.79
L3:L6	0.15	2,497.78
L6:L7	-0.18	2,215.22
L1:L7	0.40	2,202.73
L5:L6	-0.16	2,171.55
L3	0.14	1,970.01
L2:L8	0.20	1,960.51
L2:L5	-0.18	1,870.29
L5:L8	-0.16	1,430.23
L4:L6	-0.12	1,402.22

Table 8. Significant features from micro scale 2nd order logistic regression

7.2.1.2. Classification Tree Results

A classification tree model was also constructed from the training data set. The tree was grown using a minimum node size of 10 observations before attempting a split, and a minimum node size of 5 after a split. The Gini index and entropy measure were both considered as the splitting criteria, however, the Gini index was selected based on improved performance. The tree was fit using 10-fold cross validation, which provides an estimate of prediction that is used to select the best size tree. The 1-standard-error rule was used to select the tree that is within 1 standard error of the minimum cross-validated error.

The resulting tree contained 140 terminal nodes and utilized all 9 feature variables. Figure 45 shows the features used in the tree, ranked by their importance, as measured by the amount of reduction in the Gini index measure. Two of the mid-intensity level heterogeneity features (L5 and L6) were the most important variables in the tree, followed by the darker level L1 and L3 features. Once again, the L9 feature variable was the least significant in terms of Gini reduction. The classification tree produced an overall accuracy of 78%, with high precision and recall scores resulting in an F-measure of 81% (Table 9).

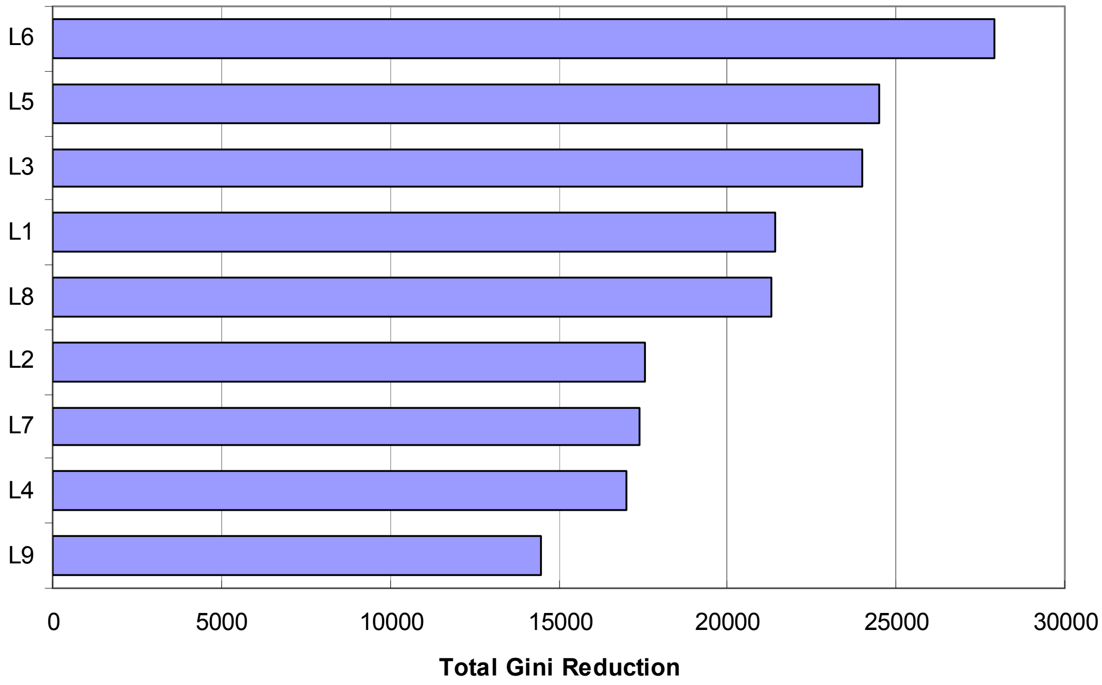


Figure 45. Micro scale feature importance from classification tree

Although the results for the single classification tree were promising on the training data, typically, trees tend to over fit the training data which results in poor predictive performance on new data sets. To address this deficiency, ensemble trees were also considered. Ensemble trees, like bagging (Breiman, 1996a), averages predictions across multiple trees produced from bootstrapped samples of the data. The ensemble tree was 2/3 of the size of the single tree, with 94 terminal nodes. Figure 46 shows the features for the ensemble tree ordered by their importance in terms of reduction in the Gini index metric. The L6 and L2 features caused the most significant decrease in the Gini index. Except for L1, the darker level heterogeneity features are typically more important than the lighter intensity levels. Although the predictive power of the ensemble tree should be improved, the classification accuracy on the training data was 26% lower than the single tree. The precision, recall, and F-measure values were also much lower (Table 9).

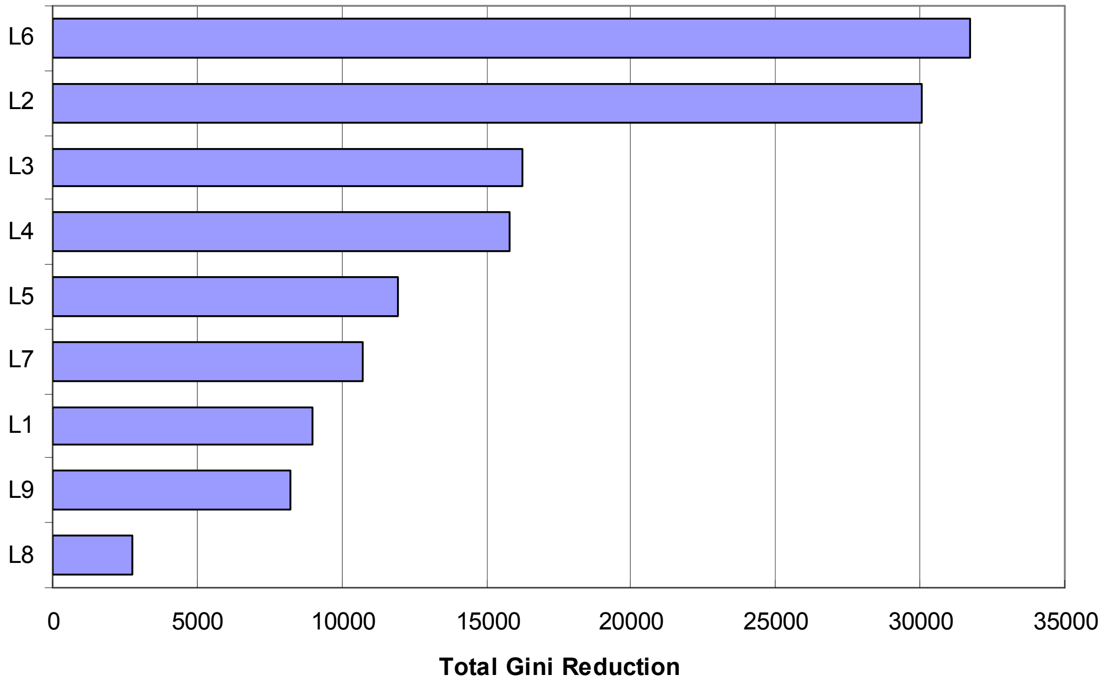


Figure 46. Micro-scale feature importance from ensemble classification tree

7.2.1.3. Neural Network Results

A neural network model was also considered as a means of capturing highly nonlinear relationships that might exist between predictor variables. The network was constructed with 2 hidden layers and 10 nodes per layer. The coefficients for the linear combinations of variables were calculated using a resilient propagation method and a convergence tolerance of 0.0001 with a maximum of 100 epochs were used to train the model. The results of the neural network classification (Table 9) show that the model classifies each observation as corroded, which produces an overall accuracy of 57%, with 100% recall and 57% precision.

7.2.1.4. Selected Models

The results of the classification of the training data for the models considered are provided in Table 9 and complete confusion matrices are given in Appendix E. In terms of overall accuracy, the single classification tree and the 2nd order logistic regression model outperformed the other methods. The F-measure results suggests the use of the single tree and the neural network models, however, the false positive rate produced by the neural network is excessive. The 1st order and 2nd order logistic regression models have similar F-measure values with the 2nd order model having better balance across the precision and recall components. The receiver operating characteristic (ROC) curve in Figure 47 shows the sensitivity vs. (1-specificity) for the models considered. The single classification tree obviously outperforms the other methods, with the 2nd order logistic regression model as the next choice. The 1st order logistic regression model and the ensemble tree produced similar values, and the neural network model offers virtually no lift from the random guess shown as the reference line in the figure.

Model	Classification Accuracy	Precision	Recall	F-Measure
Logistic Regression-1st order	57.0%	57.3%	95.9%	71.7%
Logistic Regression-2nd order	61.5%	61.9%	83.9%	71.2%
Neural Network	56.9%	56.9%	100.0%	72.5%
Classification Tree	77.8%	78.3%	84.4%	81.2%
Classification Tree-Ensemble	52.4%	60.5%	47.2%	53.0%

Table 9. Micro scale training data classification accuracy

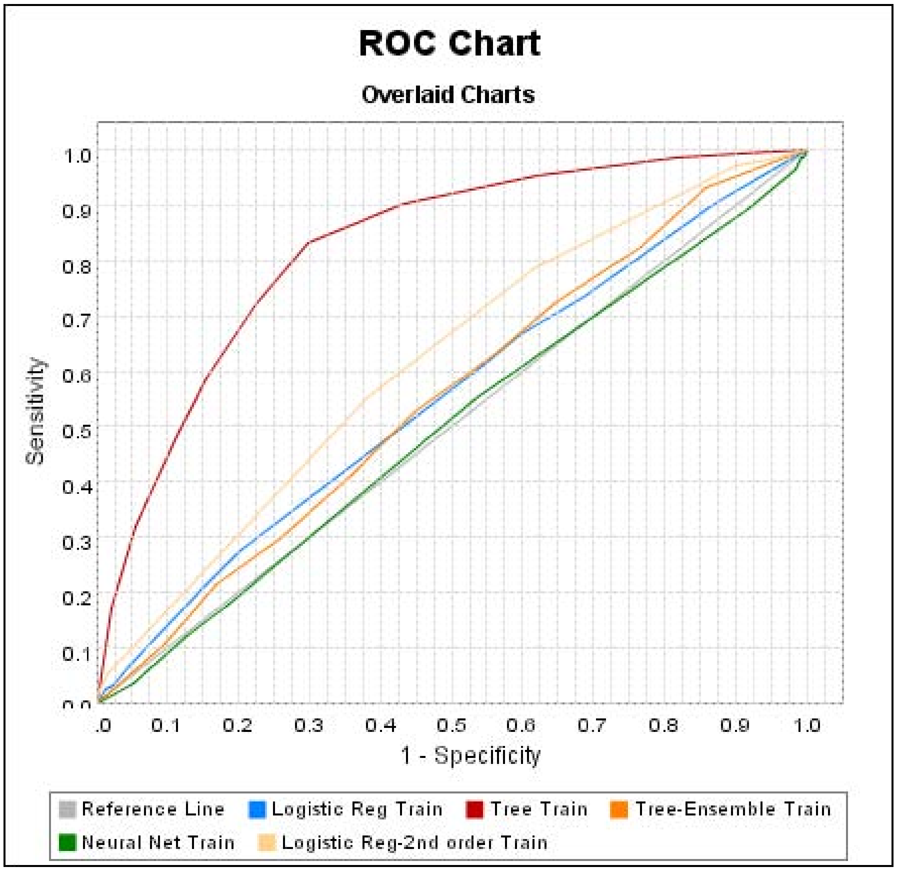


Figure 47. ROC curve for micro-scale training data

Based on the training data results, the single classification tree and the 2nd order logistic regression model would be the likely choices for model selection. However, as a final comparison, each model was used to predict the nearly 700,000 observations in the test data set. The results provided in Table 10 are almost identical to the training data results. The single tree and 2nd order logistic regression model are the best in terms of classification accuracy and the neural network continues to classify all observations as corroded which provides no discrimination of the data and a high false positive rate. Likewise, the ROC curve for the test data in Figure 48 is consistent with the training data performance. Therefore, the models selected for consideration for the growth probability function are the single classification tree and the 2nd order logistic regression model.

Model	Classification Accuracy	Precision	Recall	F-Measure
Logistic Regression	57.1%	57.4%	95.9%	71.8%
Logistic Regression-2nd order	61.5%	62.0%	83.9%	71.3%
Neural Network	57.0%	57.0%	100.0%	72.6%
Classification Tree	77.6%	78.2%	84.2%	81.1%
Classification Tree-Ensemble	52.3%	60.4%	47.1%	52.9%

Table 10. Micro scale test data classification accuracy

The selected methods were used to estimate the growth probability function over the areas in sample 1. The model estimates of the growth probability function over a given area of the sample can be depicted as a probability map like those shown in Figure 49 for area 3. The images show those regions where growth is encouraged (dark red) or inhibited (dark blue). The outline of the growth for area 3 is provided for comparison in the images.

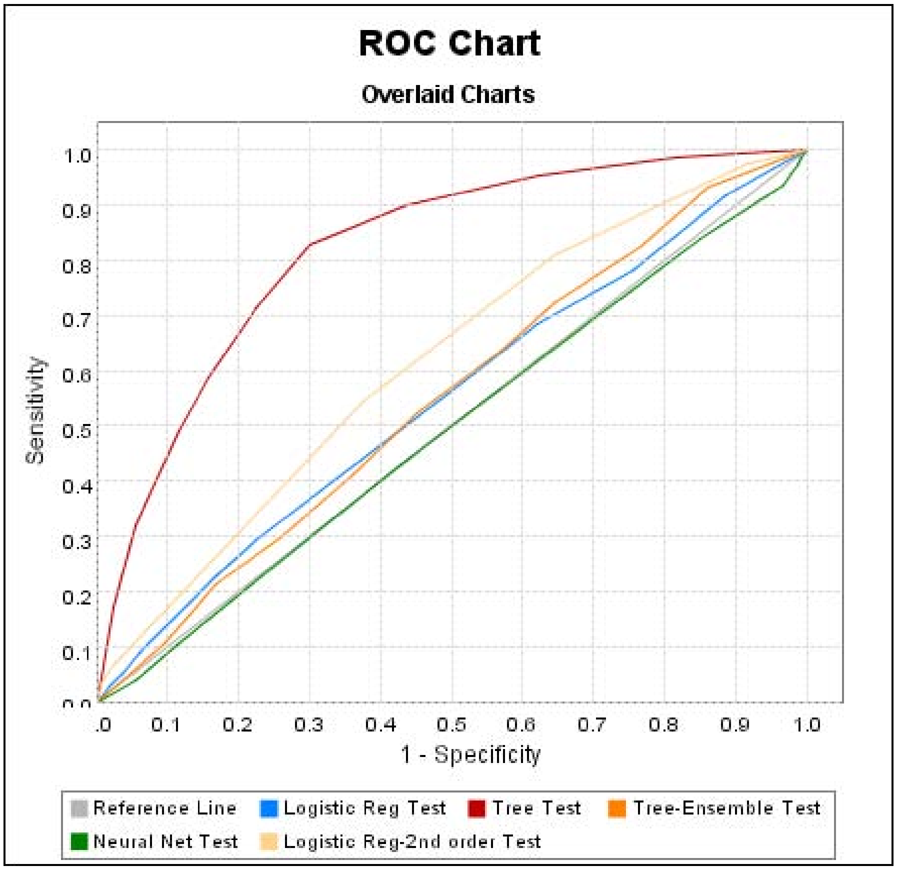


Figure 48. ROC curve for micro-scale test data

A major difference in the two methodologies is highlighted in the smoothness of the logistic regression map as opposed to the discrete surface created by the binning that occurs in classification trees. The discretized nature of the classification tree estimate is problematic in an interacting particle application because large regions will have the same probability value and cause the choice of direction to be random in those areas, with influence only along the borders of different regions.

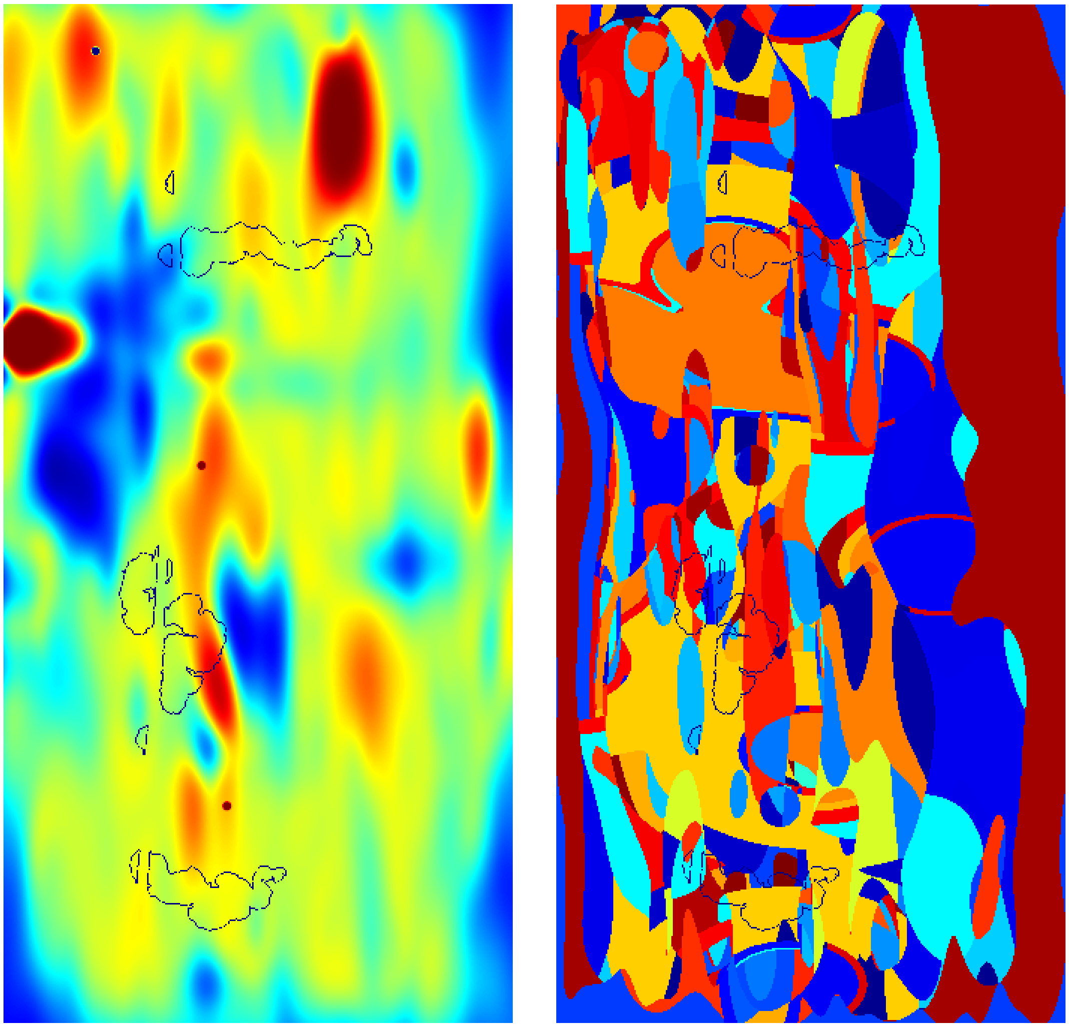


Figure 49. Growth probability maps of area 3 for 2nd order logistic regression (left) and classification tree (right) models

Assuming the growth is influenced by the heterogeneities, we would not expect dramatic differences from region to region. Instead, we would expect spatial dependence so that points in close proximity would be similar, creating a smoother surface. To address this issue, a smoothed classification tree estimate was also considered for the growth probability model. The probability map created by the classification tree was smoothed using a running mean over successive points. The smoothness of the surface is controlled by the number of points included in the running mean. The smoothed value at each location is calculated as the mean of $4\alpha + 1$ points, where α is the number of points on each side (in both dimensions) of the current location. A α value of 25 was selected for this application. Although smaller and larger values were considered $\alpha = \{10,25,50,100\}$, $\alpha = 25$ was selected because it retained most of the variations in the classification tree map while smoothing the edges enough to improve the directional control of the growth probability function. The images in Figure 50 show the classification tree and smoothed tree estimate for a region in area 3.

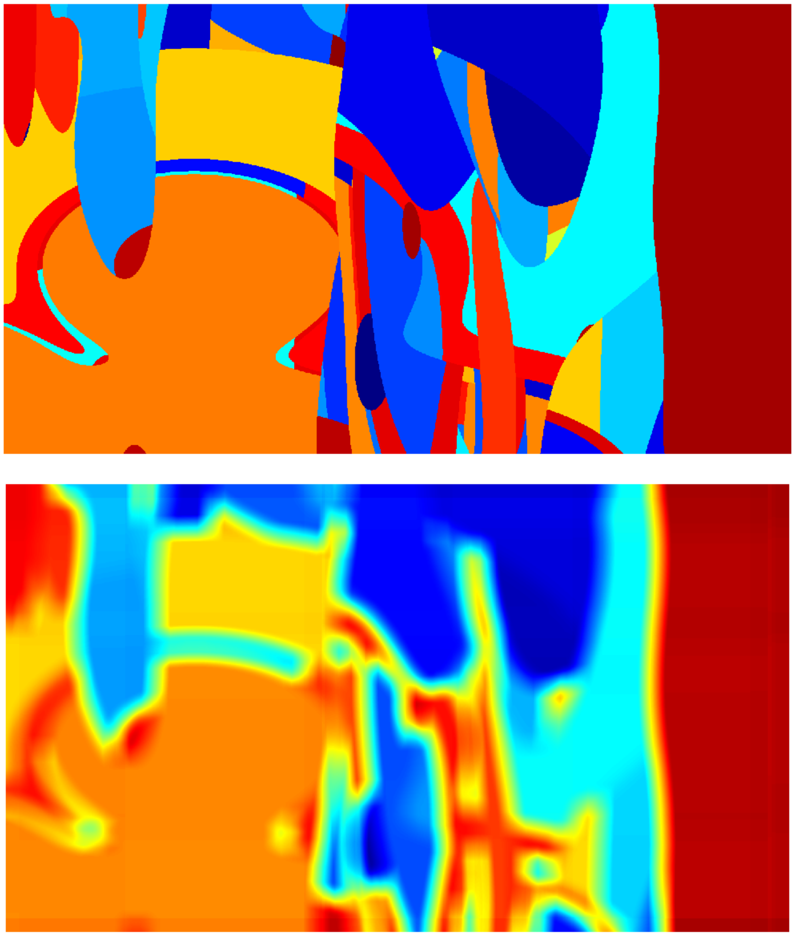


Figure 50. Classification tree (top) and smoothed tree (bottom) growth probability maps

7.2.2. Local Transition Function

According to the model definition in Chapter 3, the local transition function evolves the state of a given cell based on the states of neighboring cells and the value of the growth probability function. In interacting particle models, the local transition function is typically defined by a rule set based upon knowledge of the growth process. The rule set for this application characterizes the structure of the simulation used to predict filament growth.

7.2.2.1. Rule Set

The rule set is based upon the observed characteristics of filiform growth. At the micro-scale, we model a single filiform filament which initiates at a specified location. The direction of growth is determined by the movement of the filament head. Only noncorroded cell locations are considered which makes the filament growth self-avoiding. A variable *Range* is used to control the extent to which locations are considered for head cell movement. A *Range* value of 1 allows only the cells in the 3x3 neighborhood around the current head cell to be considered, while a *Range* of 3 considers the cells in the 7x7 neighborhood of the head cell as candidates.

Another simulation variable, *DProb*, is used to select the next location of the head cell by controlling how the growth probability values are utilized. High *DProb* values force the head cell to select the location with the highest growth probability, whereas low *DProb* values allow the distribution of growth probability values to determine the next location. After the head cell passes over a location, the remaining filament path continues to grow outward. The outward path growth is characterized by the simulation variable *Spread* which defines the width of the path by controlling the number iterations of continued growth. As explained in Chapter 5, the value of *Spread* is given by the empirical distribution of filament widths.

7.2.2.2. Simulation Algorithm

All of the variables described above are used in the filament simulation. The simulation was coded using Matlab 7.0, and variables settings of *Range*={1,2,3} and *DProb*={0.0, 0.2, 0.4, 0.6, 0.8, 1.0} were used which requires 18 design points for simulation inputs. In addition the 3 different growth probability functions described previously (2nd order logistic regression, classification tree, and smoothed classification tree) expands the dimension of the design space to create a total of 36 different simulation settings.

The simulation is run for each individual filament at each design setting, and the F-measure, precision, and recall metrics are calculated at the end of each replication. The basic algorithm for the simulation is provided below. The stopping criteria are a) there are no noncorroded locations to consider for movement (i.e. the filament has boxed itself in or run outside of the region of interest), b) the rate of corrosion has been met – as determined by the size of the filament and the desired simulation time.

1. Set the state of all cells in the region of interest =”Noncorroded”
2. A single initiation point is passed into the simulation, set the state of initiation location =”Head Cell”
3. Determine candidate locations for next Head Cell using the defined neighborhood structure and *Range* value. Only cells with state =”Noncorroded” are considered
4. Candidate selected based on growth probability and *DProb* value
5. State of selected cell =”Head Cell”, state of previous head cell =”Active”
6. Assign new active cell a *Spread* value from distribution
7. Set states of active cell neighbors =”Active”, with the same *Spread* value
8. *Spread* values of active cells are decremented by 1
9. Set state of active cell with 0 value of *Spread* =”Corroded”
10. Check for stopping conditions, repeat to step 3

The number of replications required was determined using the half-width of the confidence interval around the calculated F-measure value (Pegden *et al.*, 1995). Given the point estimate for the mean F-measure, $\overline{F}$, and half-width, h, we can calculate a confidence interval for the true F-measure value, F, such that $F = \left[\overline{F} - h, \overline{F} + h\right]$. We select a confidence level of the interval, α, so that for a random sample, $\overline{F}$ will fall within the confidence interval with probability $1-\alpha$. The half-width is calculated from the sample observations and is a measure of the precision of our estimate $\overline{F}$. Smaller values of h indicate a better estimate of F. Due to the central limit theorem, the distribution of $\overline{F}$ approaches normal as the number of replications increases, and h provides an exact $1-\alpha$ confidence interval for F. The calculation of h utilizes the upper $\alpha/2$ point of the student's t-distribution with $n-1$ degrees of freedom and the sample standard deviation of $\overline{F}$, $s\left(\overline{F}\right)$. For this application, $\alpha = 0.05$ and the simulation continues to replicate until $h < 0.001$

$$h = t_{n-1,1-\alpha/2} s\left(\overline{F}\right) \tag{28}$$

7.2.3. Interacting Particle Model Results

The simulation described above was run at each design point and each growth probability function for 10 distinct filaments in sample 1. For validation purposes, 2 filaments from a second sample were segmented and evaluated against simulated results. As a baseline for comparison, a random growth model without heterogeneity features is approximated using the same simulation with a uniform growth probability function.

7.2.3.1. Sample 1 Filaments

The median F-measure value over all simulation replications for the feature-based growth probability functions was compared to the median F-measure value for the random model replications. The chart in Figure 51 shows the mean difference in the median F-measure values across all 10 filaments in sample 1. The 3 candidate growth probability functions result in higher median F-measure values at almost every design setting. The simulated F-measure using the 2^{nd} order logistic regression model of the growth probability function is lower than the random model for low settings of *DProb*, however, when the growth is required to move in the direction of the highest growth probability, the model offers significant improvement from random.

The classification tree model also provides an improvement over the random model, although the difference in F-measure values is smaller than the other models'. In contrast, the classification tree performs best at low *DProb* values when the distribution of growth probabilities is used to determine a cell's likelihood to transition to a head cell state. The performance degrades as *DProb* is increased. This is due to the discretized nature of the classification tree growth probability function. For example, when the filament nears the boundary between low and high growth probability values and *DProb* is 1, then the filament must move across the boundary. When the classification tree values are smoothed, the performance of the simulation is improved and the relationship with the *DProb* variable follows that of the smooth estimate generated by the 2^{nd} order logistic regression estimate. The simulated F-measure for the smooth tree model is generally as good as or better than the 2^{nd} order logistic regression model

The results suggest higher *DProb* values as the best design setting for the smooth tree and logistic regression models. The best choice for *Range* is not readily determined from the results. Both the smooth tree and logistic regression have similar performance across the values of *Range*. The results in Figure 51 are based on the median F-measure from the simulation replications which provides a measure of central tendency without the influence of extreme values. The use of the mean F-measure value (Figure 52), however, provides results similar to the median. The results for each individual filament are provided in Appendix G.

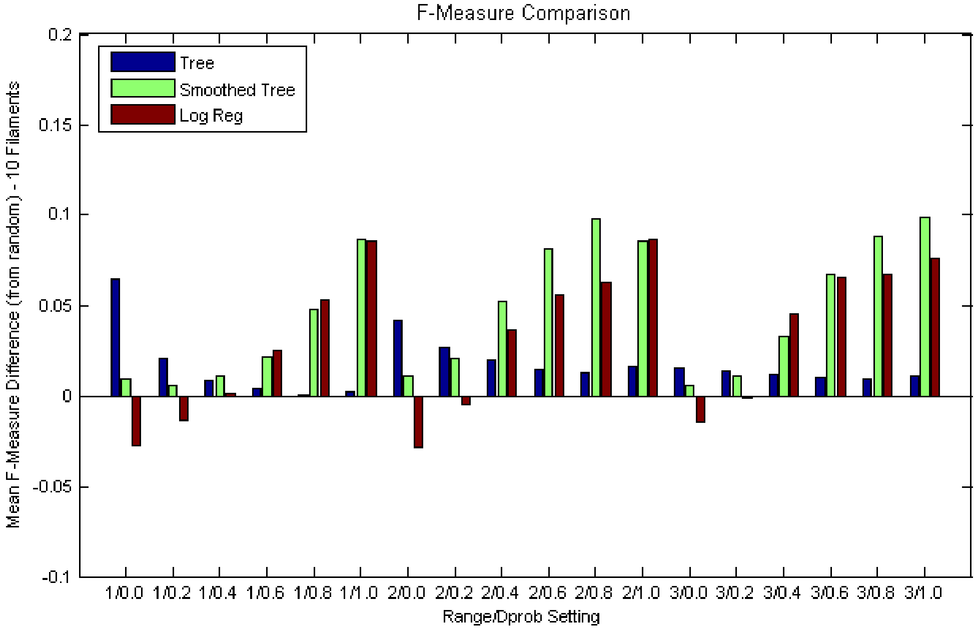


Figure 51. Average difference from random model for mean F-measure

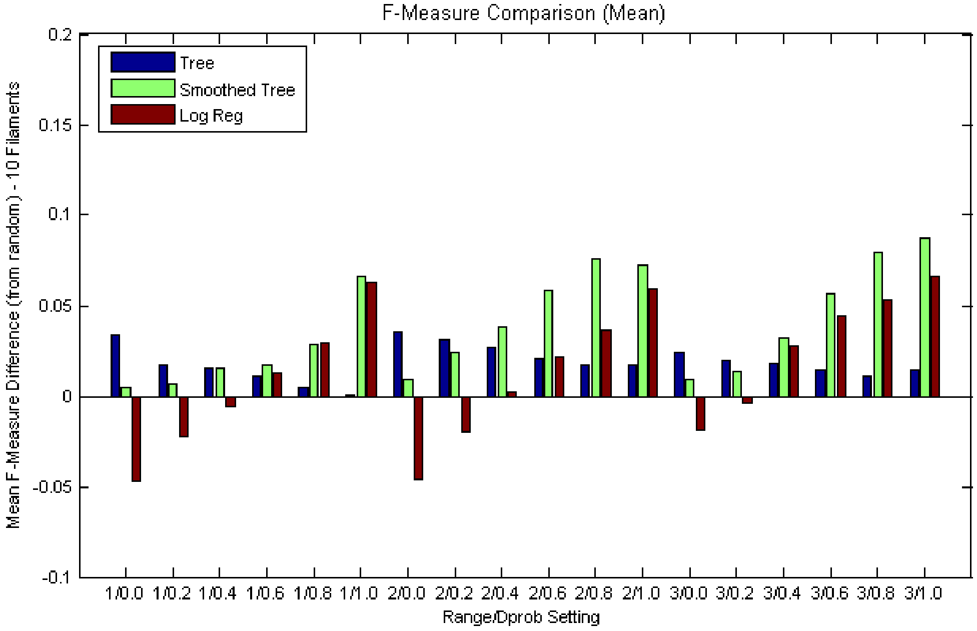


Figure 52. Average difference from random model for median F-measure

The statistical significance of the differences in F-measure values can be tested using the Wilcoxon rank-sum test. The Wilcoxon rank-sum test is designed to test the hypothesis that samples from two independent continuous populations have equal medians. For each area we combine the replication F-measure values at a given design setting for the feature-based and random models. The values are ranked and the sums of the ranks for each population are then compared. If the medians are similar, then we would expect the sums to be nearly equal for both samples. For large samples, the rank sum is approximately normal, and we can compare the test statistic to the critical value from the normal distribution.

The chart in Figure 53 shows the proportion of filaments (out of 10) where the respective feature-based model produces F-measure values greater than the random model at a 95% level of significance, according to the Wilcoxon rank-sum test. The results show that for *DProb*=1 and *Range* of 1 or 2, the 2nd order logistic regression model outperforms the random model in the simulation of every filament. The smooth tree model is also statistically better than random for the majority of filaments when high values of *DProb* are used.

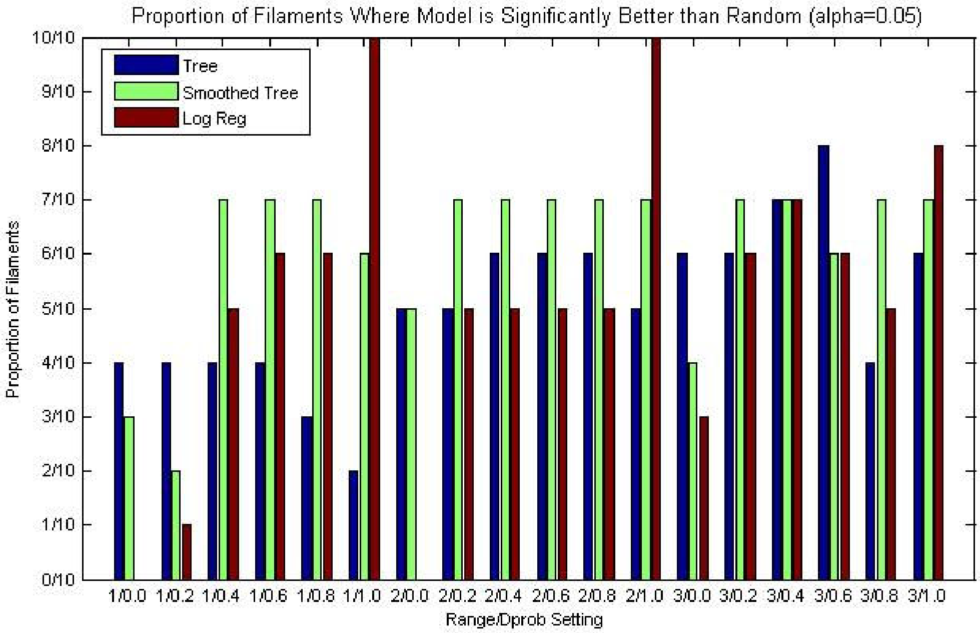


Figure 53. Proportion of filaments where feature-based model is significantly better than random

7.2.3.2. Sample 2 Filaments

The results for the filaments in sample 1 show that the feature-based models offer an improvement upon a model of random growth. To validate these results, filaments from a second sample were segmented and simulated with the feature-based and random models. The experimental design for sample 2 was identical to that of sample 1. All preparation, imaging, and data collection methods were repeated.

Despite the similarities, the growth in sample 2 was noticeably different from sample 1. The image in Figure 54 shows one of the areas in sample 2 after 89 hours of exposure. The amount of growth on the sample is much greater than the growth shown in the images for sample 1. This difference is due to the significant increase in the number of initiated filaments in sample 2. It is possible that the amount of initiation in sample 2 is related to the greater delamination around the scribe that is evident in the images. However, we assume the amount of initiation does not affect the growth of individual filaments beyond the possible interaction that can occur when close filaments limit the available direction of growth.

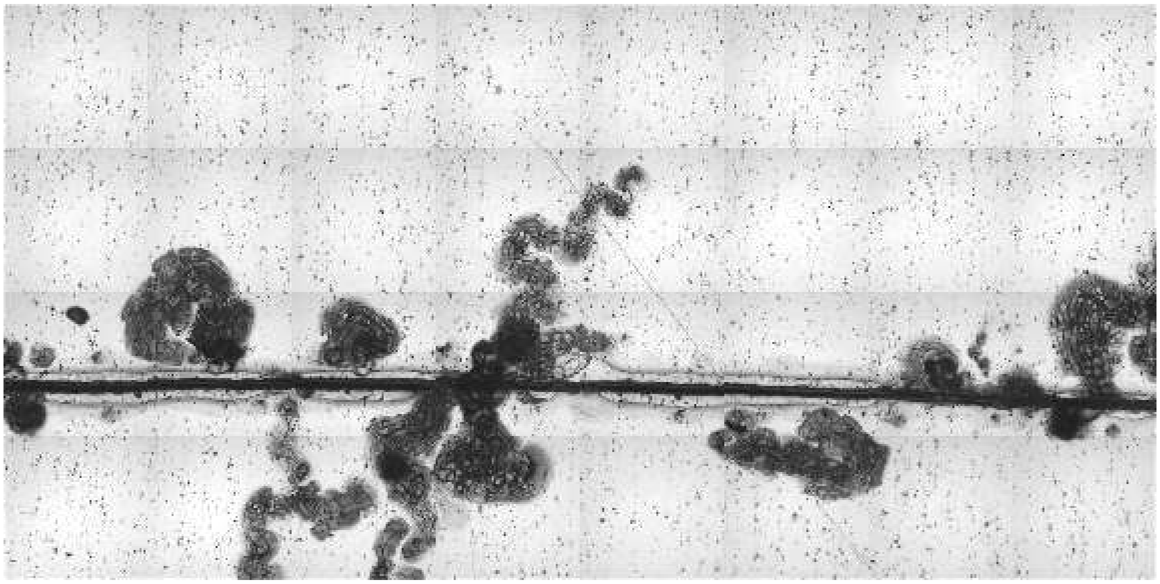


Figure 54. Image of sample 2 area 4, t=89 hours

To validate the sample 1 results, two filaments were selected at random from area 4 in sample 2. The only influence on the selection of the filaments was the ease of segmentation to capture distinct filaments without the influence of neighboring filaments. Figure 55 shows the images at t=89 for the two filaments selected from sample 2. The feature values were determined and the growth probability function estimates were calculated. The filaments were then simulated at the previous design points using the interacting particle model.

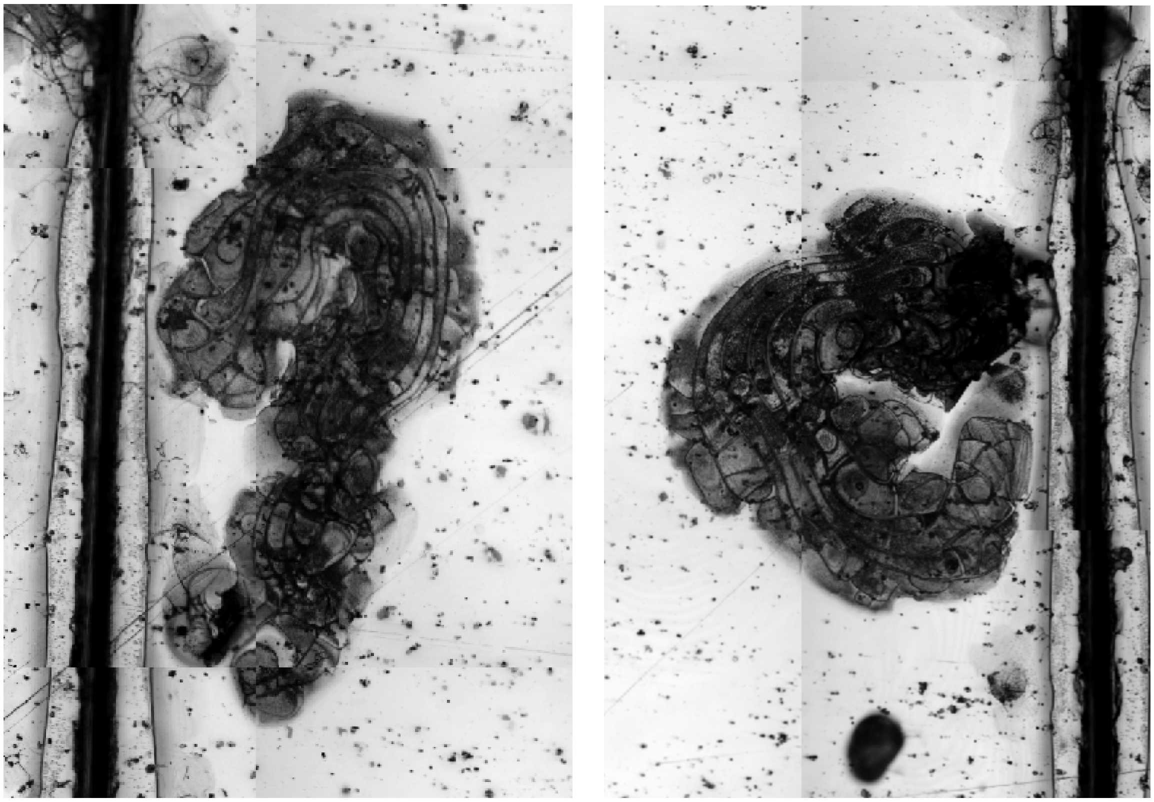


Figure 55. CLSM images of Sample 2 filaments (filament 1:left, filament 2:right)

The interacting particle model results for filament 1 of sample 2 are provided in Figure 56. The results from the logistic regression model show significant improvement over the random growth model especially for high *DProb* values. The smooth tree model is also better than random for *DProb*=1. Like the other models, the median F-measure for the classification tree follows the same pattern from the sample 1 results. The tree results are better than random at low values of *DProb* and then degrade as *DProb* increases. The logistic regression results are clearly better than the other models and significantly better than the results from the random model.

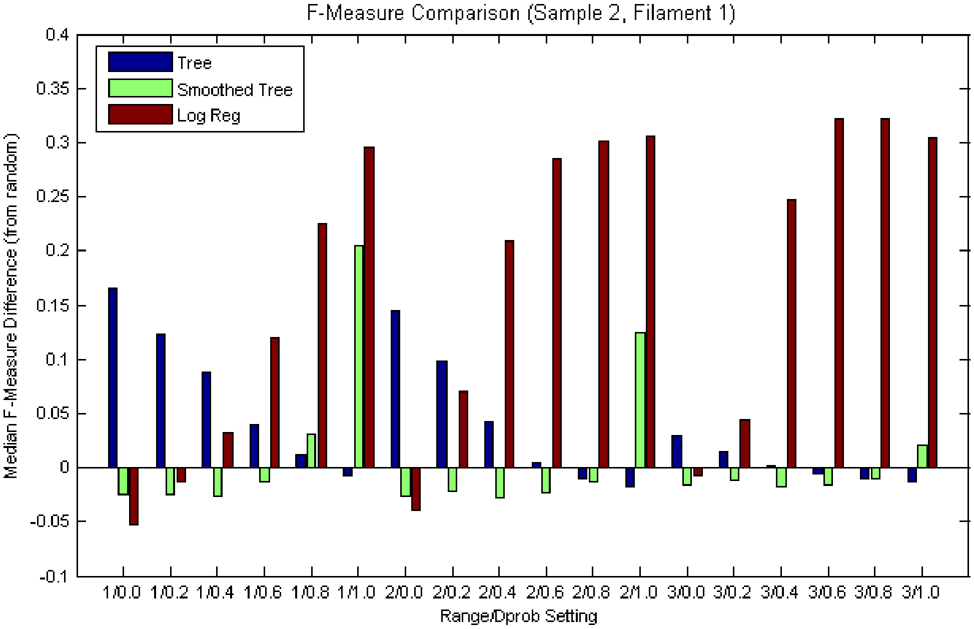


Figure 56. Sample 2, filament 1, F-measure difference from random

The results for the 2nd filament from sample 2 are not as promising as the previous filament. Figure 57 shows the median F-measure difference from the random model for filament 2. Once again, the classification tree results improve as *Dprob* is decreased, while the results for the smooth tree and logistic regression improve as *DProb* increases. In fact, *DProb*=1 is the only setting where models perform better than random, with the logistic regression and smooth tree having median F-measure values better than random at all 3 *Range* settings. Despite the lower results for most models and design settings, if we average the difference in median F-measure values across both filaments, the results would be consistent with the results from the sample 1 filaments. The selection of the 2nd order logistic regression model for the growth probability function and a *Range* of 1 or 2, with a *DProb* setting of 1.0 produced results better than the random model across all filaments in sample 1 and sample 2.

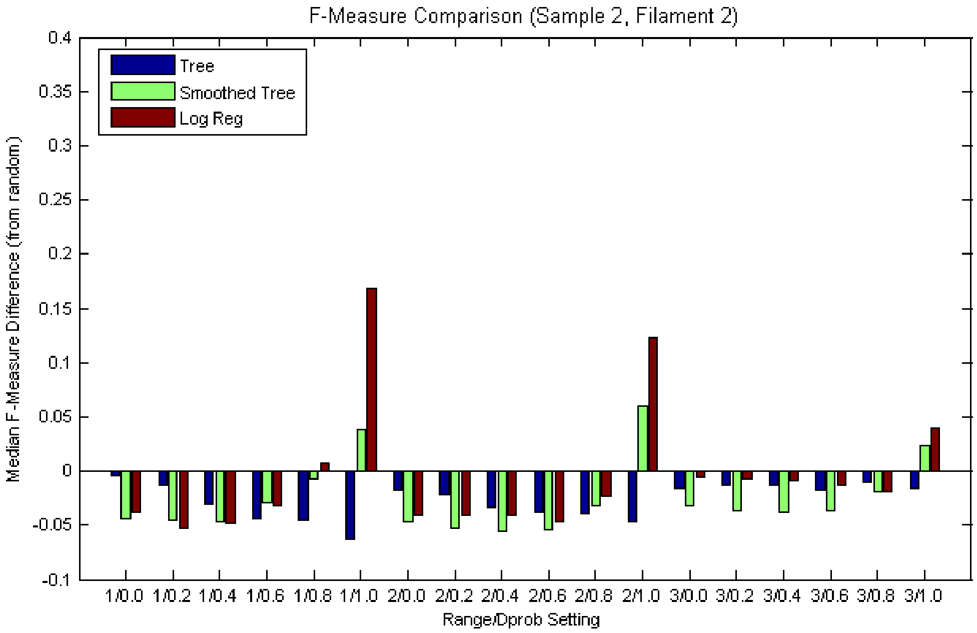


Figure 57. Sample 2, filament 2, F-measure difference from random

CHAPTER 8. CONCLUSIONS AND RECOMMENDATIONS

The final chapter summarizes the research presented in this dissertation. The conclusions from the results in the previous chapter are presented along with recommendations for future research.

8.1. Summary

This dissertation has been focused on the space-time prediction of random events and growth. Typically, the approach to these problems only utilizes the geographic space information. The goal of this work was to incorporate feature space analysis into the space-time prediction of event initiation and growth. The inclusion of the feature information allows the models to better describe the initiation and growth processes, and therefore improve the predictive abilities of the models. The model is able to account for relationships that exist between the feature variables and event initiation and growth that are likely not known or understood. In this respect, this research represents an improvement in the prediction of space-time initiation and growth as well as in corrosion modeling. As opposed to most corrosion models which focus on a single descriptive metric, this research offers a true space-time prediction for corrosion growth. Specifically, this research has generated the following contributions.

- The primary goal of this research was to formulate a spatiotemporal model that incorporates feature space analysis in space-time event initiation and growth prediction. The feature-based point prediction model (Liu, 1999), was extended to a feature-based, space-time model for growth prediction. A feature-based interacting particle model was developed for growth prediction, and combined with the feature-based point prediction model to capture the initiation and growth of a complex system.
- The properties of the growth model have also been presented. It has been shown that if we assume that the features included in the model are significant to the

process, then we are guaranteed that the error or deviance in the model fit will be less than or equal to the error or deviance from model that does not include these features. This result can be seen for a wide range of model classes for the growth probability function included in the proposed interacting particle model.

- The application of the model was demonstrated with data of filiform corrosion growth in aluminum. Data was collected from macro and micro scale images of filiform growth on samples of AA2024-T3. The features included in the application were those derived directly from macro scale images, and the density of various intensity heterogeneities in the micro scale images.
- The effectiveness of the proposed feature-based model in terms of error reduction and prediction accuracy has also been presented. The initiation model showed significant improvement in the majority of areas in terms of the likelihood of occurrence over the uniform spatial transition density. Likewise, the growth model showed a significant improvement over the random growth model for all filaments considered in this research.

8.2. Conclusions

The application of the proposed model has provided a great deal of valuable insight not only for the process of filiform initiation and growth, but also to the use of this type of modeling for similar applications. The following conclusions and observations are provided based on the results for the initiation and growth models in both the macro and micro scale approach.

- The results from the macro scale approach show that geometric features derived from the digital images are insufficient to describe the growth process. These results imply that features beyond what can be described by the filiform growth itself influence the process.

- The micro scale approach suggests that the heterogeneities visible in CLSM images can be used to predict filament initiation and growth. Incorporating this information into initiation and growth models can improve the prediction accuracy over that of random spatial models that do not account for these relationships.
- In general, the heterogeneity features offer improvement over the uniform spatial density for initiation, however, there were areas when the model results were not better than the random spatial density. This result is likely due to other features that can be identified in the CLSM images, such as the width of the scribe or delamination of the coating around the scribe.
- The heterogeneities with intensities associated with the L3 feature variable offer the most significant influence over the initiation process. In general, the darker features provide the best results for prediction of initiation location.
- The analysis of the growth of segmented filaments has shown that a constant growth rate is a reasonable assumption. After initiation, the amount of new growth that occurs varies somewhat from filament to filament, but on average is constant and independent of the time of exposure.
- Interacting particle models can provide an acceptable representation of filament growth over time. The basic known growth characteristics can be captured in the transition functions which define the interacting particle model. Realizations of the model produce predicted filament growth that exhibits similar patterns and growth behavior as actual filaments over a given region of material.
- The results from the growth model predictions show that the density of heterogeneities with various intensity levels can be used to predict filament growth. The inclusion of these features offers a significant improvement in the prediction of growth over other random models which do not account for this

information. This result is consistent with the theoretical properties of the growth probability function approximation.

- Although the relationship of the growth process with the heterogeneity features appears highly nonlinear, a larger influence of the darker heterogeneities on filament growth has been observed.
- Classification tree models of the growth probability function can capture highly nonlinear relationships with feature variables and provide the best fit of the data, however, the predictive ability of the model is degraded with new data sets.
- The discretized probability values generated by the classification tree can degrade the performance of the proposed interacting particle model, and a smoothed function over the geographic region appears to improve results.

8.3. Suggestions for Future Research

This research represents an initial endeavor to incorporate feature space analysis into the prediction of growth for complex systems. Specifically, this work offers an initial introduction to actual space-time prediction of corrosion evolution. In addition to the conclusions drawn, the results of the research have generated further focus areas which warrant future investigation.

- The location of filament initiation could not be accurately identified in the CLSM images in the micro scale approach. Despite additional experiments, the centroid location of growth at the initial imaging time had to be used as the assumed initiation location. Obviously, an improvement upon this approximation should improve the model outcome and help to validate the conclusion of the influence of heterogeneities on initiation.
- The difference in the amount of initiation from samples 1 and 2 and the corresponding difference in the width of the scribe on the samples should be investigated. The width of the scribe at a given location could easily be

incorporated into the existing model and additional scribe features could dramatically improve the predictive results for the initiation model.

- In addition to the scribe and coating features, other environmental variables such as temperature and humidity can be used to account for differences in initiation numbers.
- The significance of the L3 heterogeneities in the prediction of initiation location needs to be analyzed further. A better chemical classification of these heterogeneities could provide further insight into the relationship with initiation location. This result is also applicable to the growth model where complex relationships appear to exist with the darker heterogeneities influencing the growth of filaments. These types of heterogeneities need to be better classified in order to better understand the process.
- More complex models should be considered for approximating the growth probability function. The computational considerations drove the selection of models for this research. If feasible with the extremely large data sets, the use of generalized additive models or MARS should be considered and could offer an improvement over the current methods.
- The rule set which governs the interacting particle model used in this research was developed based on simple understood characteristics of filament growth. Given other research available on filiform corrosion, the model can likely be improved upon by capturing further known growth behavior. Undoubtedly, more sophisticated methods exist for implementing the interacting particle model which could greatly reduce computational time and vastly improve the ability to perform additional tests.
- The current method utilizes the empirical distribution of filament widths to determine the *Spread* value for the filament path. The growth probability

function can also be used to determine the likelihood that the path continues to spread into neighboring cells.

- The interaction between filaments during growth should also be investigated. The current method assumes the only interaction among filaments is the limited space for future growth caused by another path occupying those locations. It is possible, although not observed in this data, that a dependence exists between multiple filaments growing in the same region.

Appendix A: Image Registration and Segmentation

Image Registration

Formally, we select a set of points in the base image $\{(x_i, y_i): i = 1, \ldots, N\}$ and the corresponding set of points in the input image $\{(x'_i, y'_i): i = 1, \ldots, N\}$. The key to registration is to determine the function $f(x, y)$ which can be separated into components $f_x(x, y)$ and $f_y(x, y)$ such that $x'_i = f_x(x_i, y_i)$ and $y'_i = f_y(x_i, y_i)$.

The images used in this research are of metallic samples placed in stages at consecutive time periods. Because of the rigid samples and templates used for camera position, the transformation needed for this process must be capable of correcting scale, rotation, and translation differences in time-sequenced images. Therefore, the spatial transformation in the images can be represented as a linear conformal transformation, or a transformation of the Cartesian coordinate system. The linear conformal transformation is represented by the following component functions of the base image reference points.

$$\begin{aligned} x' &= \alpha x \cos(\theta) - \alpha y \sin(\theta) + \beta_x \\ y' &= \alpha x \sin(\theta) - \alpha y \cos(\theta) + \beta_y \end{aligned} \tag{29}$$

The scaling parameter, α, allows for differences in the scale of the two images. Scaling differences can be caused by differences in the focal length used to create the digital images. For example, if the focal length in the input image is slightly larger than the focal length of the base image, the objects in the resulting image will appear larger due to the increase in magnification. The rotational parameter, θ, adjusts for differences in the rotation of the camera from the horizontal axis between the base and input image. The translation parameter, β, represents the shift in location in the x and y direction from the base and input image.

Image Segmentation: thresholding

The purpose of thresholding is to select a single pixel value, or threshold, in order to separate objects of interest from the image background. Depending on the image, objects are typically lighter (darker) than their background, which implies that pixels above (below) the threshold value make up the objects of interest. Specifically, thresholding seeks to determine the value *T* for image *X* which produces the binary image *Y*.

$$Y(i,j)=\begin{cases}0 & \text{if } X(i,j)\leq T\\ 255 & \text{if } X(i,j)>T\end{cases} \tag{30}$$

The resulting binary image *Y* will have objects represented by white pixels and background pixels will be black. Because thresholding utilizes pixel intensity, images are typically in a grayscale format. However, methods for color thresholding are similar. Thresholding is an attractive segmentation method because it results in a significant reduction in data storage and the binary image is much simpler to analyze (Wolfram, 2004).

There are many methods that can be used to threshold an image. Perhaps the most obvious is to utilize the histogram of pixel values (Pal and Pal, 1993). The histogram represents the probability distribution of pixel values in the image. The effectiveness of any thresholding technique depends on the distribution of the pixels in the histogram. The presumption is that the conditional probability distribution for pixel values that belong to the object will be shifted from the conditional probability distribution of pixel values for the background. Figure 58 shows an example of pixel distributions. The intersection of the two conditional probability distributions provides the value of threshold that should be used to classify pixels. Assuming equal costs, histogram thresholding results in a minimum Bayes risk classification. The decision rule for pixels in image *X*, can be expressed as a function of the prior probability of the pixel in H_j, which can be estimated empirically.:

$$\phi(X)=\underset{j=1,2}{\arg\max}\, P_j f_{X|H_j}(X\mid H_j)$$

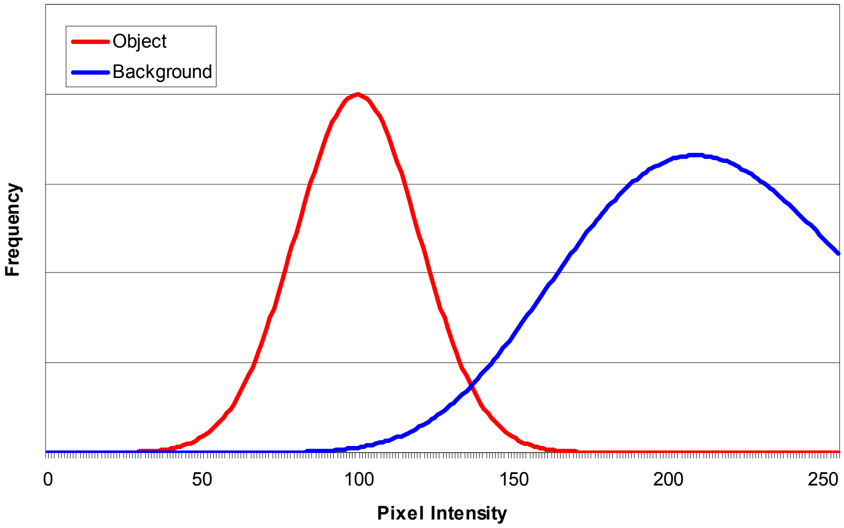


Figure 58. Example of histogram thresholding

In most cases, the conditional probability distributions for objects and background are not observable in a histogram. Typically, only the distribution of pixel values of the entire image can be constructed. Because of the variation that can occur from image to image, a constant threshold value would not be effective. In order to threshold the image, an adaptive method must be utilized. One possible heuristic selects the threshold value that equals the mean of the pixel values above the threshold averaged with the mean of the pixel values below the threshold. Other methods seek to optimize a specific error metric. One popular methodology is called the Otsu method (Otsu, 1979).

The Otsu method selects the threshold that minimizes the variance within the two resulting classes. Suppose an image contains N pixels, and we have differentiated M different levels of pixel intensity, so that n_i is the number of pixels at intensity level $i \in M$. We wish to separate the pixel values into two classes: C_1 which has pixels with intensity levels $[1,\ldots,k]$, and C_2 which has pixels with intensity levels $[k+1, \ldots, M]$. The variance within each class for a threshold $t \in M$ is defined as $\sigma_W^2(t) = q_1(t)\sigma_1^2(t) + q_2(t)\sigma_2^2(t)$ where

$$q_1(t) = \sum_{i=0}^{t-1} P(i)$$

$$q_2(t) = \sum_{i=t}^{M} P(i)$$

$$\sigma_1^2(t) \equiv \text{variance of pixels in } C_1$$

$$\sigma_2^2(t) \equiv \text{variance of pixels in } C_2$$

$$P(i) = \frac{n_i}{N}$$

However, an easier method exists which utilizes the relationship of the between-class variance and the within-class variance. The between-class variance is defined as

$$\begin{aligned}\sigma_B^2(t) &= \sigma^2 - \sigma_W^2(t)\\ &= q_1(t)[\mu_1(t) - \mu]^2 + q_2(t)[\mu_2(t) - \mu]^2\end{aligned}$$

where μ is the combined mean of the two classes, and σ^2 is the combined variance. We can substitute using $\mu = q_1\mu_1 + q_2\mu_2$. The resulting formula for between-class variance becomes $\sigma_B^2 = q_1(t)q_2(t)[\mu_1(t) - \mu_2(t)]^2$.

Variance is a measure of self-similarity within groups. The within-class variance averages over both classes. When a threshold is too high, the variance for *C*$_1$ will be too high and too many pixels will be classified as *C*$_1$. When the threshold is too low, the variance for *C*$_2$ will be too high and too many pixels will be classified as *C*$_2$. Therefore, the optimal threshold maximizes the between-class variance, and thus minimizes the within-class variance. The threshold value can be iteratively selected by calculating σ_B^2 using the means of each resulting cluster and the number of pixels in each cluster.

Image Segmentation: edge detection

Another method of image segmentation is called edge detection. If we assume that objects in an image are characterized by continuity, then pixels that are close in proximity should have color values that are also close. A pixel with a sharp change in color or intensity can be called an edge. The desire is that these edges match up with the natural boundary of the objects within the image.

Many methods are available for detecting edges in images. These methods are generally categorized as gradient, which use first derivative information or Laplacian, which uses 2nd derivative information. The gradient method finds edges by calculating the maximum and minimum in the first derivative of the image. Laplacian methods find the zero values in the second derivative of the image. To allow for the calculation of the gradients at each pixel location, convolution masks, or kernels, are used. The gradient is equal to the value of the pixel convolved with the kernel.

It would be possible to use a classification method, such as k-nearest neighbor, to classify each pixel based on its RGB intensity value, similar to the thresholding method. However, there are portions of the rivet, particularly the hole in the center, which have dramatically different color values than the rest of the rivet. Therefore, only portions of the rivet would be classified as such. Edge detection, however, can be used to group the entire rivet object. For the purposes of this project, edge detection will only be used after the image has been segmented using thresholding. Therefore, edge detection will be performed on the resulting binary image to categorize the foreground pixels into objects. A foreground pixel *P* is labeled an edge point if:

$$\begin{array}{|c|c|c|}\hline P_1 & P_2 & P_3 \\ \hline P_4 & P & P_5 \\ \hline P_6 & P_7 & P_8 \\ \hline \end{array} \cdot \begin{array}{|c|c|c|}\hline 0 & -1 & 0 \\ \hline -1 & 4 & -1 \\ \hline 0 & -1 & 0 \\ \hline \end{array} \neq 0$$

Other convolution kernels can be used to represent different neighborhood structures. Changing the kernel primarily affects the smoothness of the resulting edge.

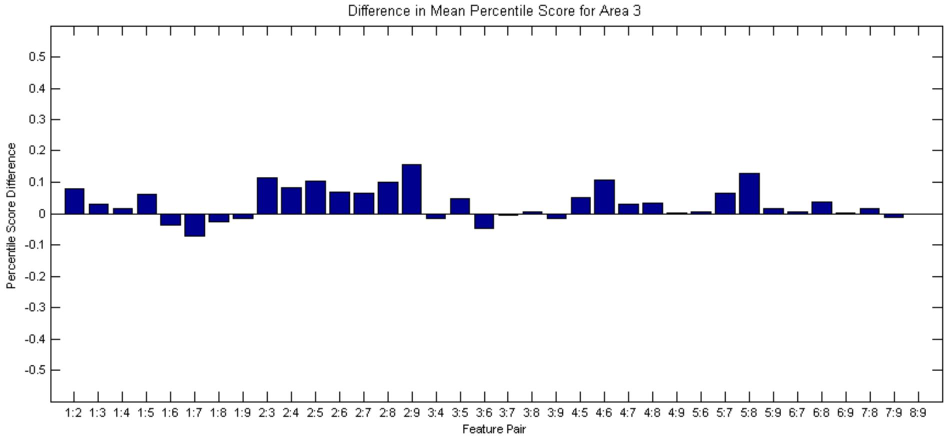
Appendix B: Macro Scale Correlation Matrix

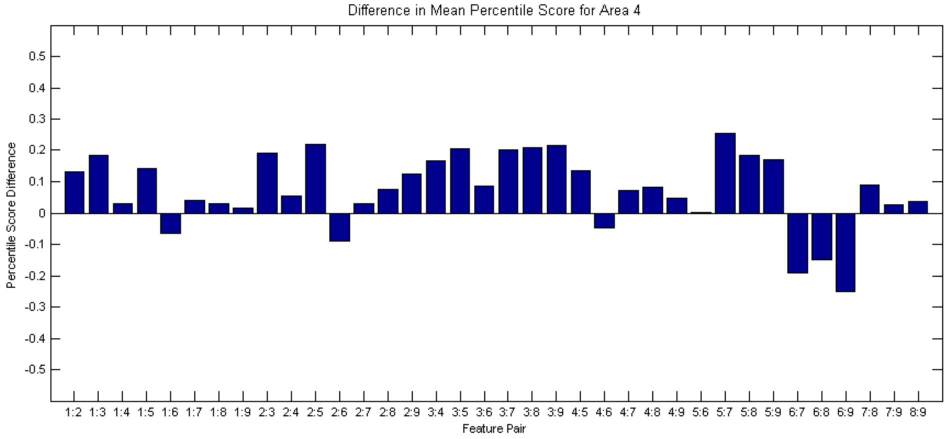
	init_d	anginit	ageinit	samesd	Corrode	scribe_d	corr_d	numcorr_3	numcorr_5	numcorr_9	numcorr_17	sizecorr	solidcorr	extregcorr	pc_ang	ci_ang	ci_d	left	right	gap	dang_ratio	gapsc	sameside
init_d	1.00	0.01	0.06	0.14	-0.32	0.99	0.99	-0.12	-0.15	-0.20	-0.28	0.12	-0.09	0.18	0.00	0.00	0.14	0.41	0.37	0.32	0.46	0.99	0.14
anginit		1.00	-0.02	0.04	-0.02	0.01	0.02	0.00	0.00	0.00	-0.01	0.01	-0.01	0.00	0.01	0.01	-0.02	0.01	0.04	0.01	0.00	0.01	0.04
ageinit			1.00	0.43	0.02	0.08	-0.01	0.02	0.03	0.04	0.06	0.11	-0.11	0.17	-0.06	-0.09	0.12	-0.01	0.01	0.01	-0.23	0.07	0.43
samesd				1.00	-0.04	0.20	0.11	0.00	0.00	0.01	0.02	0.09	-0.11	0.14	-0.03	-0.03	0.16	0.13	0.14	0.14	-0.25	0.20	1.00
Corrode					1.00	-0.31	-0.32	0.29	0.37	0.48	0.58	0.01	0.00	0.05	0.03	0.02	0.03	-0.52	-0.48	-0.48	-0.19	-0.32	-0.04
scribe_d						1.00	0.99	-0.11	-0.15	-0.20	-0.27	0.12	-0.09	0.19	0.00	0.00	0.15	0.43	0.40	0.34	0.44	1.00	0.20
corr_d							1.00	-0.12	-0.16	-0.22	-0.29	0.05	-0.02	0.09	-0.01	0.00	0.04	0.41	0.38	0.32	0.52	0.98	0.11
numcorr_3								1.00	0.91	0.73	0.50	0.02	-0.02	0.03	-0.02	0.00	0.02	-0.25	-0.26	-0.30	-0.07	-0.12	0.00
numcorr_5									1.00	0.90	0.65	0.03	-0.03	0.04	-0.02	0.00	0.02	-0.33	-0.34	-0.38	-0.10	-0.16	0.00
numcorr_9										1.00	0.85	0.04	-0.04	0.06	-0.01	0.00	0.04	-0.43	-0.44	-0.47	-0.13	-0.22	0.01
numcorr_17											1.00	0.07	-0.07	0.10	-0.01	0.00	0.07	-0.53	-0.51	-0.51	-0.18	-0.29	0.02
sizecorr												1.00	-0.86	0.70	0.08	0.08	0.60	0.05	-0.01	0.02	-0.24	0.12	0.09
solidcorr													1.00	-0.59	-0.11	-0.13	-0.55	-0.04	0.02	-0.02	0.23	-0.09	-0.11
extregcorr														1.00	0.00	0.00	0.76	0.04	0.04	0.03	-0.35	0.19	0.14
pc_ang															1.00	0.59	0.06	0.08	-0.07	0.00	-0.02	0.00	-0.03
ci_ang																1.00	0.05	0.05	-0.02	0.01	-0.02	0.00	-0.03
ci_d																	1.00	0.03	0.01	0.02	-0.44	0.14	0.16
left																		1.00	0.68	0.81	0.23	0.45	0.13
right																			1.00	0.85	0.22	0.42	0.14
gap																				1.00	0.18	0.37	0.14
dang_ratio																					1.00	0.44	-0.25
gapsc																						1.00	0.20
sameside																							1.00

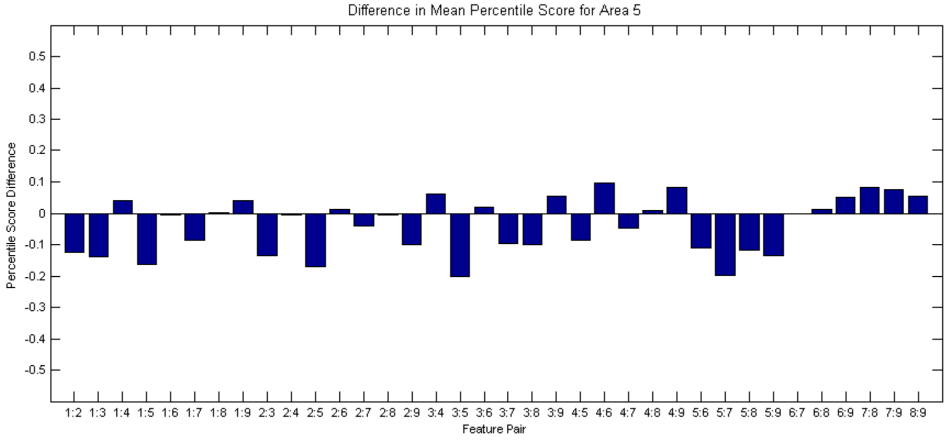
Appendix C: Macro-Scale Logistic Regression Output

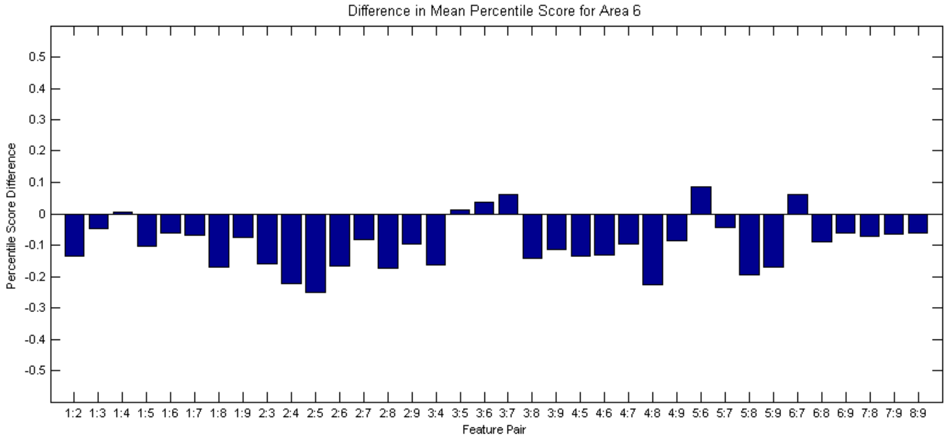
Variable	Estimate	Std.Err.	t-Statistic	Pr(|t|)
(Intercept)	2.16	0.12	18.06	0
init_d	-0.12	2.87E-03	-42.01	0
anginit	-0.1	0.01	-10.96	6.30E-28
ageinit	-0.3	0.04	-7.06	1.68E-12
scribe_d	0.05	0.01	7.71	1.27E-14
corr_d	-0.08	0.01	-13.58	5.26E-42
numcorr_17	0.01	5.01E-04	15.19	0
pc_ang	1.13	0.09	13.15	0
ci_ang	0.22	0.08	2.83	4.71E-03
ci_d	0.16	0.01	18.89	0
left	0	2.29E-05	-5.83	5.65E-09
right	0	2.48E-05	-4.32	1.59E-05
gap	2.10E-04	1.04E-04	2.01	0.04
dang_ratio	0.11	0.02	5.13	2.86E-07
gapsc	-1.56E-05	3.95E-06	-3.94	8.17E-05
sameside	-0.56	0.04	-13.56	7.29E-42
sizecorr	0	7.66E-05	-12.55	3.81E-36
dang_ratio:sameside	-0.03	0.01	-4.1	4.11E-05
pc_ang:ci_ang	-0.48	0.04	-11.89	1.40E-32
pc_ang:ci_d	-0.11	0.01	-15.47	5.29E-54
pc_ang:gap	0	8.40E-05	-6.8	1.04E-11
pc_ang:dang_ratio	-0.14	0.01	-9.41	5.01E-21
ci_ang:ci_d	-0.04	0.01	-5.95	2.74E-09
ci_ang:gap	0	6.93E-05	-6.58	4.60E-11
ci_ang:dang_ratio	-0.15	0.01	-10.4	2.38E-25
ci_d:gap	-7.69E-05	9.78E-06	-7.86	3.77E-15
ci_d:dang_ratio	-0.08	0.01	-14.03	9.97E-45
gap:dang_ratio	6.52E-06	1.55E-05	0.42	0.67
pc_ang:ci_ang:ci_d	0.05	3.50E-03	14.34	0
pc_ang:ci_ang:gap	4.60E-04	3.73E-05	12.32	0
pc_ang:ci_ang:dang_ratio	0.09	0.01	12.64	0
pc_ang:ci_d:gap	1.26E-04	7.49E-06	16.87	0
pc_ang:ci_d:dang_ratio	0.01	2.81E-03	3.19	1.41E-03
pc_ang:gap:dang_ratio	4.05E-06	1.07E-05	0.38	0.7
ci_ang:ci_d:gap	1.09E-04	6.14E-06	17.7	0
ci_ang:ci_d:dang_ratio	0.01	1.49E-03	7.52	5.55E-14
ci_ang:gap:dang_ratio	-7.68E-07	4.80E-06	-0.16	0.87
ci_d:gap:dang_ratio	-1.32E-05	2.24E-06	-5.9	3.68E-09
pc_ang:ci_ang:ci_d:gap	-8.58E-05	3.49E-06	-24.61	9.82E-134
corr_d:sameside	0	3.09E-03	-1.48	0.14
corr_d:sizecorr	5.19E-05	5.56E-06	9.33	0
sameside:sizecorr	3.88E-04	7.35E-05	5.28	1.30E-07
corr_d:sameside:sizecorr	-1.58E-05	5.45E-06	-2.89	3.83E-03

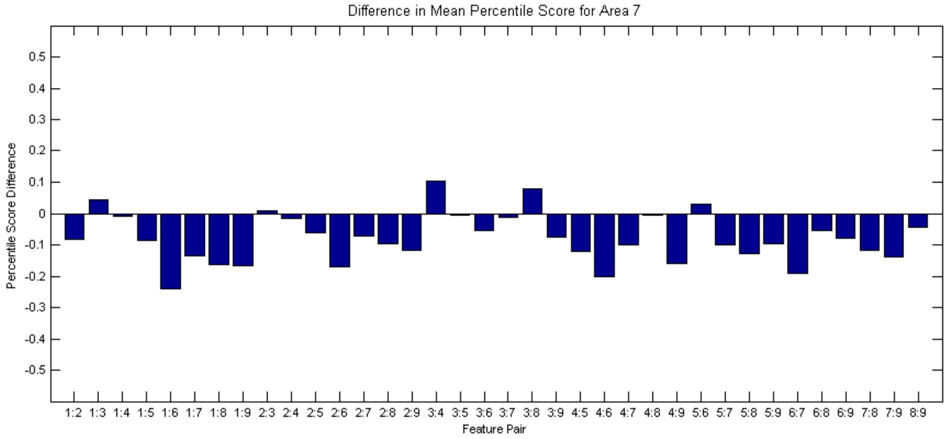
Appendix D: Initiation Percentile Scores by Area

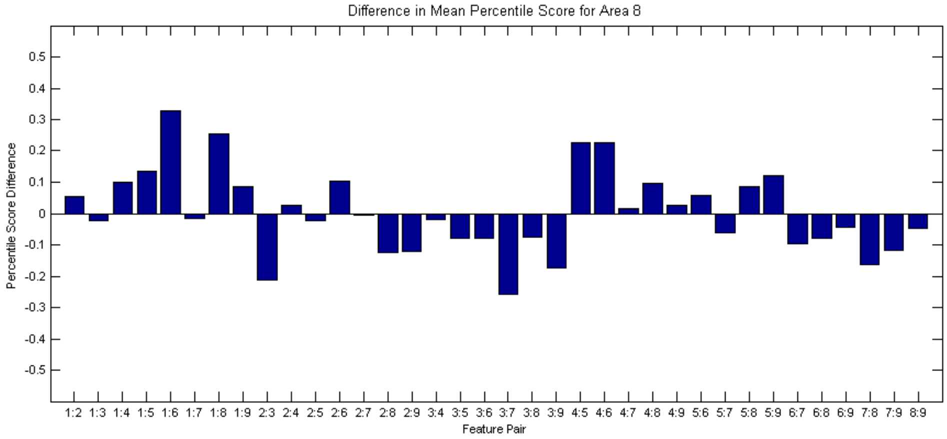

Difference in Mean Percentile Score for Area 3

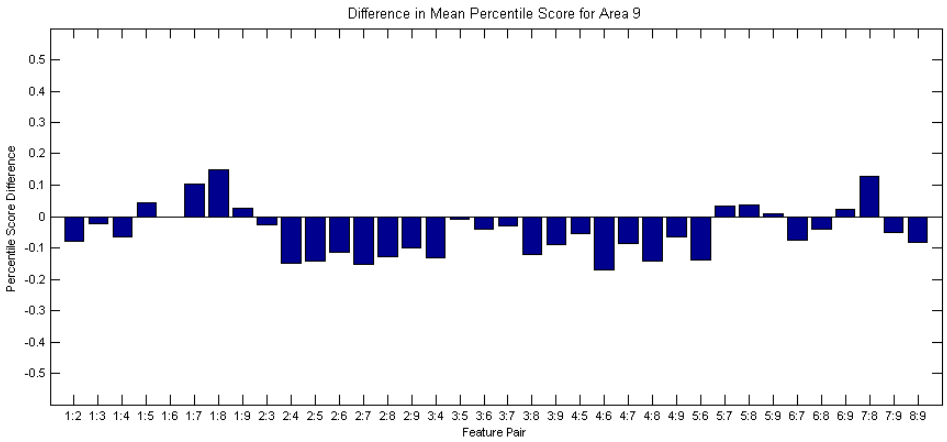

Difference in Mean Percentile Score for Area 4

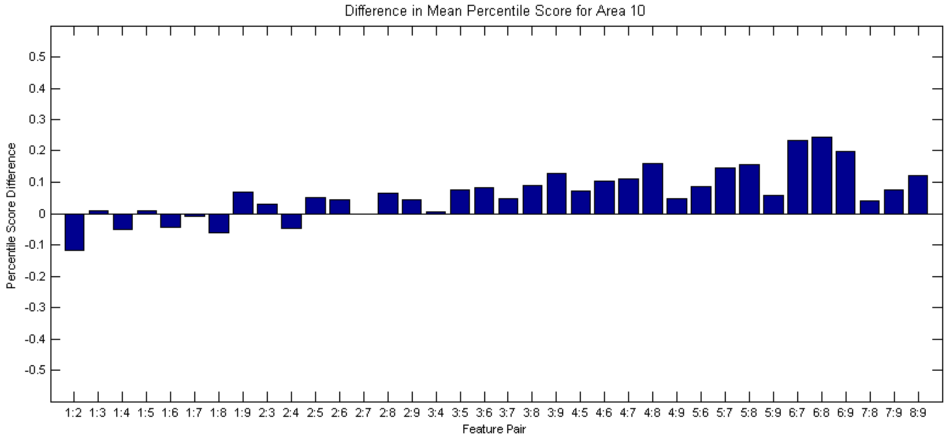

Difference in Mean Percentile Score for Area 5

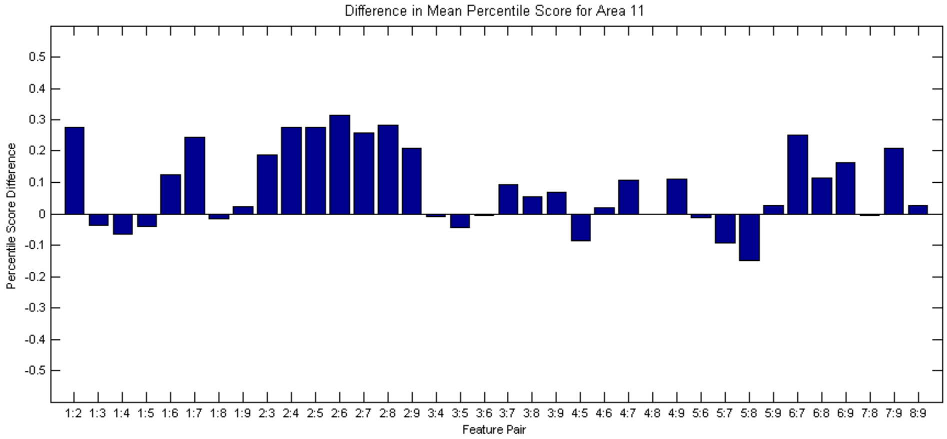

Difference in Mean Percentile Score for Area 6
Percentile Score Difference
0.5
0.4
0.3
0.2
0.1
0
-0.1
-0.2
-0.3
-0.4
-0.5
1:2 1:3 1:4 1:5 1:6 1:7 1:8 1:9 2:3 2:4 2:5 2:6 2:7 2:8 2:9 3:4 3:5 3:6 3:7 3:8 3:9 4:5 4:6 4:7 4:8 4:9 5:6 5:7 5:8 5:9 6:7 6:8 6:9 7:8 7:9 8:9
Feature Pair


Difference in Mean Percentile Score for Area 7
Percentile Score Difference
0.5
0.4
0.3
0.2
0.1
0
-0.1
-0.2
-0.3
-0.4
-0.5
1:2 1:3 1:4 1:5 1:6 1:7 1:8 1:9 2:3 2:4 2:5 2:6 2:7 2:8 2:9 3:4 3:5 3:6 3:7 3:8 3:9 4:5 4:6 4:7 4:8 4:9 5:6 5:7 5:8 5:9 6:7 6:8 6:9 7:8 7:9 8:9
Feature Pair


Difference in Mean Percentile Score for Area 8
Percentile Score Difference
0.5
0.4
0.3
0.2
0.1
0
-0.1
-0.2
-0.3
-0.4
-0.5
1:2 1:3 1:4 1:5 1:6 1:7 1:8 1:9 2:3 2:4 2:5 2:6 2:7 2:8 2:9 3:4 3:5 3:6 3:7 3:8 3:9 4:5 4:6 4:7 4:8 4:9 5:6 5:7 5:8 5:9 6:7 6:8 6:9 7:8 7:9 8:9
Feature Pair


Difference in Mean Percentile Score for Area 9
Percentile Score Difference
0.5
0.4
0.3
0.2
0.1
0
-0.1
-0.2
-0.3
-0.4
-0.5
1:2 1:3 1:4 1:5 1:6 1:7 1:8 1:9 2:3 2:4 2:5 2:6 2:7 2:8 2:9 3:4 3:5 3:6 3:7 3:8 3:9 4:5 4:6 4:7 4:8 4:9 5:6 5:7 5:8 5:9 6:7 6:8 6:9 7:8 7:9 8:9
Feature Pair


Difference in Mean Percentile Score for Area 10
Percentile Score Difference
0.5
0.4
0.3
0.2
0.1
0
-0.1
-0.2
-0.3
-0.4
-0.5
1:2 1:3 1:4 1:5 1:6 1:7 1:8 1:9 2:3 2:4 2:5 2:6 2:7 2:8 2:9 3:4 3:5 3:6 3:7 3:8 3:9 4:5 4:6 4:7 4:8 4:9 5:6 5:7 5:8 5:9 6:7 6:8 6:9 7:8 7:9 8:9
Feature Pair


Difference in Mean Percentile Score for Area 11
Percentile Score Difference
0.5
0.4
0.3
0.2
0.1
0
-0.1
-0.2
-0.3
-0.4
-0.5
1:2 1:3 1:4 1:5 1:6 1:7 1:8 1:9 2:3 2:4 2:5 2:6 2:7 2:8 2:9 3:4 3:5 3:6 3:7 3:8 3:9 4:5 4:6 4:7 4:8 4:9 5:6 5:7 5:8 5:9 6:7 6:8 6:9 7:8 7:9 8:9
Feature Pair

Appendix E: Micro-Scale Confusion Matrices

		Truth		
		0	1	Total
Predicted	0	37,394	34,690	72,084
	1	602,414	808,620	1,411,034
	Total	639,808	843,310	1,483,118

		Truth		
		0	1	Total
Predicted	0	16,029	14,938	30,967
	1	257,586	347,069	604,655
	Total	273,615	362,007	635,622

1st order Logistic Regression (training:top, test:bottom)

		Truth		
		0	1	Total
Predicted	0	203,922	135,623	339,545
	1	435,886	707,687	1,143,573
	Total	639808	843,310	1,483,118

		Truth		
		0	1	Total
Predicted	0	87,087	58,190	145,277
	1	186,528	303,817	490,345
	Total	273,615	362,007	635,622

2nd order Logistic Regression (training:top, test:bottom)

		Truth		
		0	1	Total
Predicted	0	442,162	131,784	573,946
	1	197,646	711,526	909,172
	Total	639808	843,310	1,483,118

		Truth		
		0	1	Total
Predicted	0	188,439	57,132	245,571
	1	85,176	304,875	390,051
	Total	273,615	362,007	635,622

Classification Tree (training:top, test:bottom)

		Truth		
		0	1	Total
Predicted	0	379,513	445,176	824,689
	1	260,295	398,134	658,429
	Total	639808	843,310	1,483,118

		Truth		
		0	1	Total
Predicted	0	161,772	191,430	353,202
	1	111,843	170,577	282,420
	Total	273,615	362,007	635,622

Ensemble Tree (training:top, test:bottom)

		Truth		
		0	1	Total
Predicted	0	0	0	0
	1	639,808	843,310	1,483,118
	Total	639808	843,310	1,483,118

		Truth		
		0	1	Total
Predicted	0	0	0	0
	1	273,615	362,007	635,622
	Total	273,615	362,007	635,622

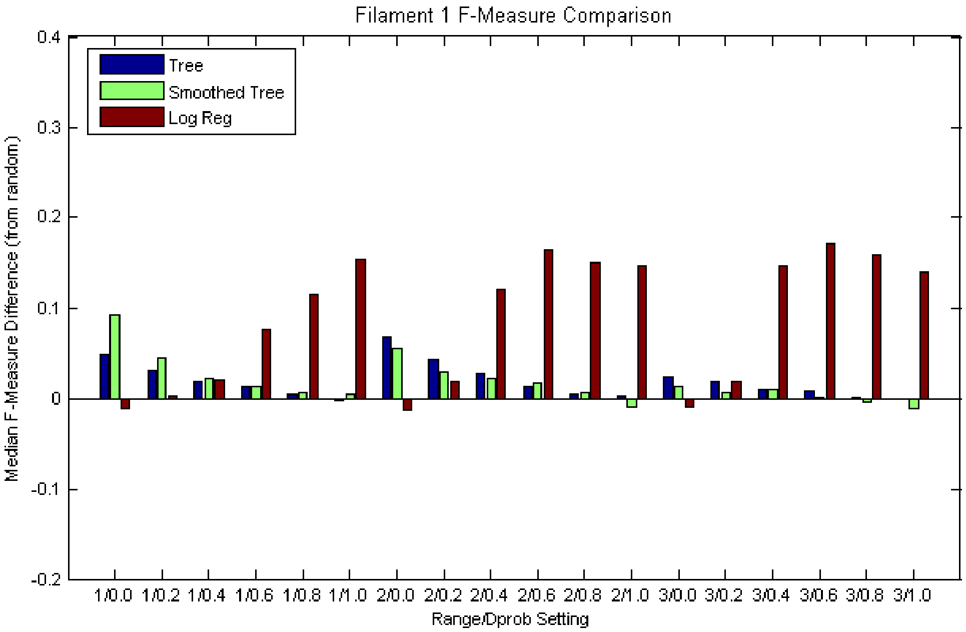
Neural Network (training:top, test:bottom)

Appendix F: Micro-Scale Logistic Regression (2nd Order) Output

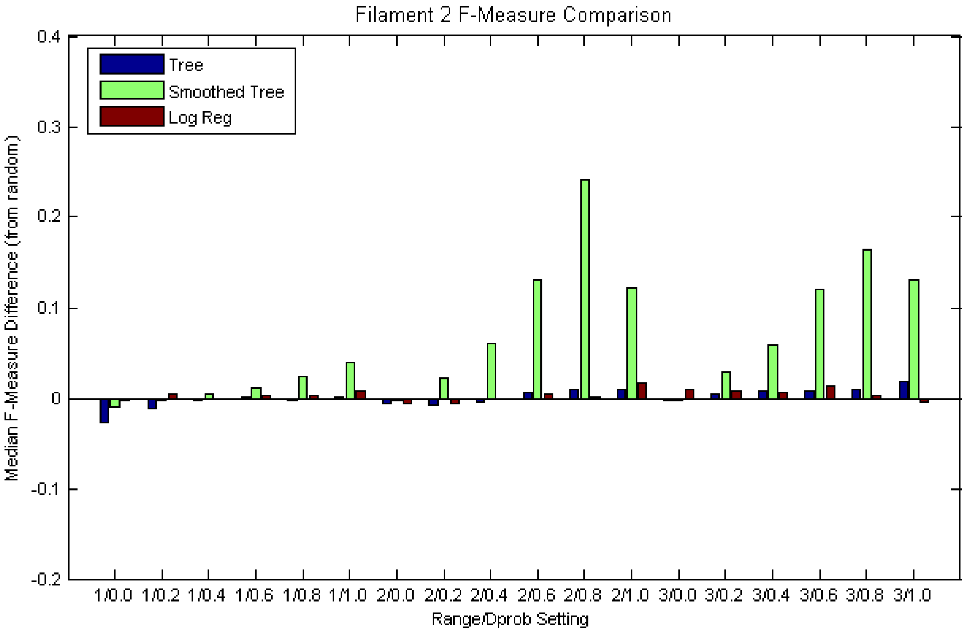
Variable	Estimate	Std.Err.	t-Statistic	Pr(|t|)
(Intercept)	0.26	4.34E-03	60.9	0
L1	-0.4	0.01	-76.18	0
L2	0.01	3.15E-03	3.94	8.09E-05
L3	0.14	3.17E-03	44.38	0
L4	0.03	3.37E-03	10	0
L5	-0.02	3.14E-03	-5.83	5.58E-09
L6	0.19	2.55E-03	72.54	0
L7	0.04	4.97E-03	7.51	6.11E-14
L8	-0.09	0.01	-14.07	6.05E-45
L9	0.17	0.03	5.69	1.29E-08
L1:L1	0.06	7.55E-04	83.38	0
L2:L2	-0.16	2.42E-03	-64.42	0
L3:L3	0.13	2.24E-03	55.96	0
L4:L4	-0.08	2.18E-03	-34.83	9.23E-266
L5:L5	-0.07	2.49E-03	-28.02	1.01E-172
L6:L6	-0.06	1.88E-03	-30.37	1.40E-202
L7:L7	0.05	3.96E-03	11.45	0
L8:L8	-0.1	3.79E-03	-26.39	1.82E-153
L9:L9	0.01	5.13E-04	12.28	0
L1:L2	0.17	3.09E-03	53.87	0
L1:L3	-0.08	4.49E-03	-17.71	3.38E-70
L1:L4	0.18	3.59E-03	51.14	0
L1:L5	-0.03	4.33E-03	-6.36	2.07E-10
L1:L6	0.14	3.94E-03	35.63	0
L1:L7	0.4	0.01	46.93	0
L1:L8	-0.62	0.01	-67.37	0
L1:L9	-0.44	0.04	-11.17	5.77E-29
L2:L3	0.09	3.08E-03	30.73	0
L2:L4	-0.14	3.75E-03	-37.14	9.54E-302
L2:L5	-0.18	4.08E-03	-43.25	0
L2:L6	-0.09	2.83E-03	-31.14	8.21E-213
L2:L7	-0.33	0.01	-65.88	0
L2:L8	0.2	4.54E-03	44.28	0
L2:L9	-0.69	0.03	-21.17	1.78E-99
L3:L4	0.06	3.26E-03	18.66	0
L3:L5	0.19	3.63E-03	51.99	0
L3:L6	0.15	3.10E-03	49.98	0
L3:L7	0.05	4.25E-03	10.6	0
L3:L8	0.04	0.01	6.06	1.34E-09
L3:L9	0.57	0.02	23.57	0
L4:L5	-0.02	3.82E-03	-4.81	1.48E-06
L4:L6	-0.12	3.19E-03	-37.45	9.64E-307
L4:L7	-0.05	0.01	-9.18	4.50E-20

L4:L8	0.13	0.01	23.87	0
L4:L9	-0.8	0.04	-21.17	2.02E-99
L5:L6	-0.16	3.34E-03	-46.6	0
L5:L7	0.27	4.19E-03	64.82	0
L5:L8	-0.16	4.19E-03	-37.82	0
L5:L9	0.68	0.03	25.04	0
L6:L7	-0.18	3.91E-03	-47.07	0
L6:L8	0.11	3.90E-03	29.48	0
L6:L9	-0.17	0.01	-14.52	9.46E-48
L7:L8	0.29	0.01	36.04	0
L7:L9	-1.15	0.05	-21.78	3.49E-105
L8:L9	1.33	0.05	29.26	0

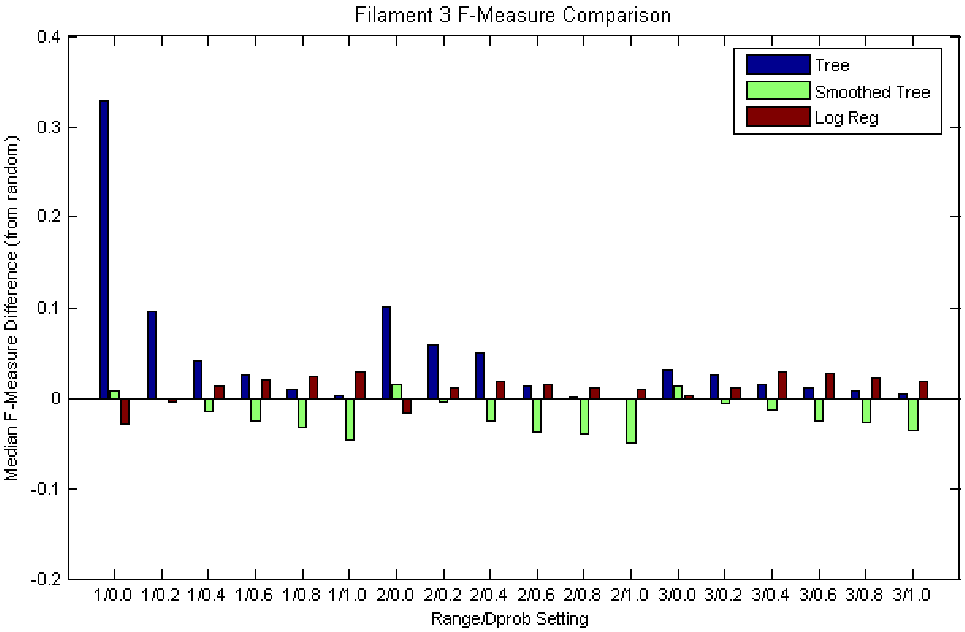
Appendix G: F-Measure Comparison for Sample 1 Filaments

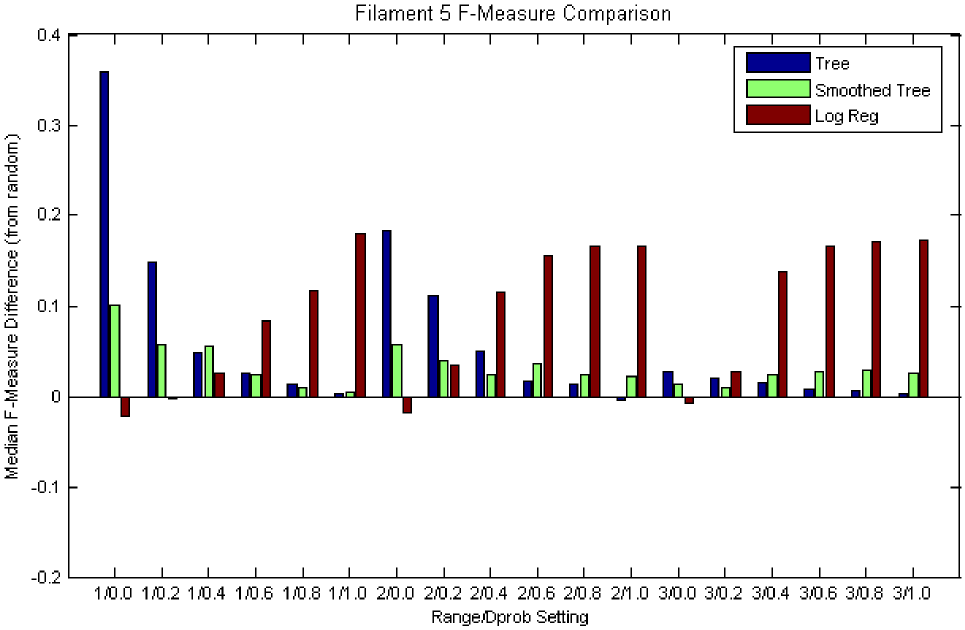


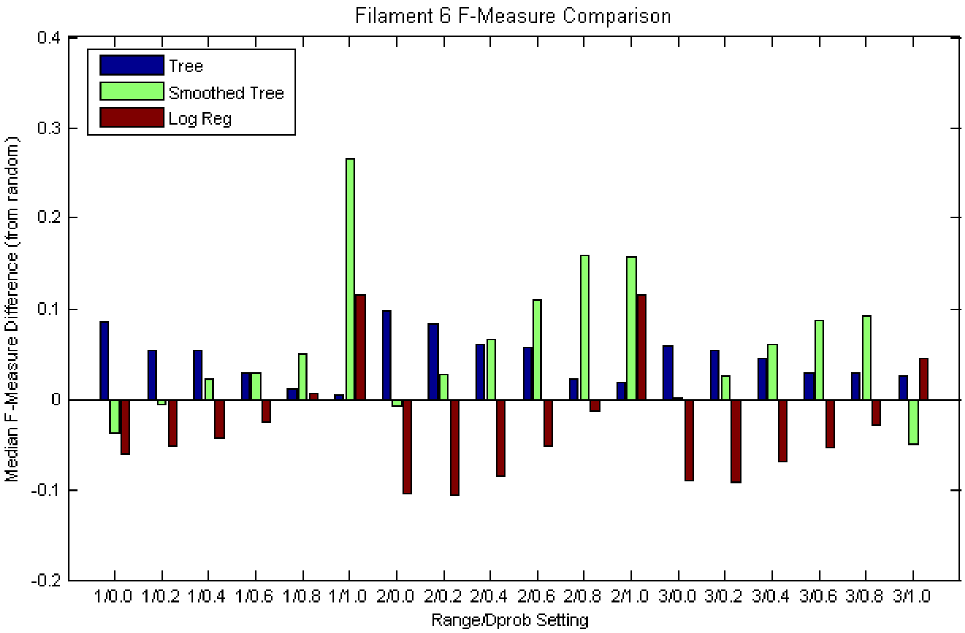
Filament 1 F-Measure Comparison

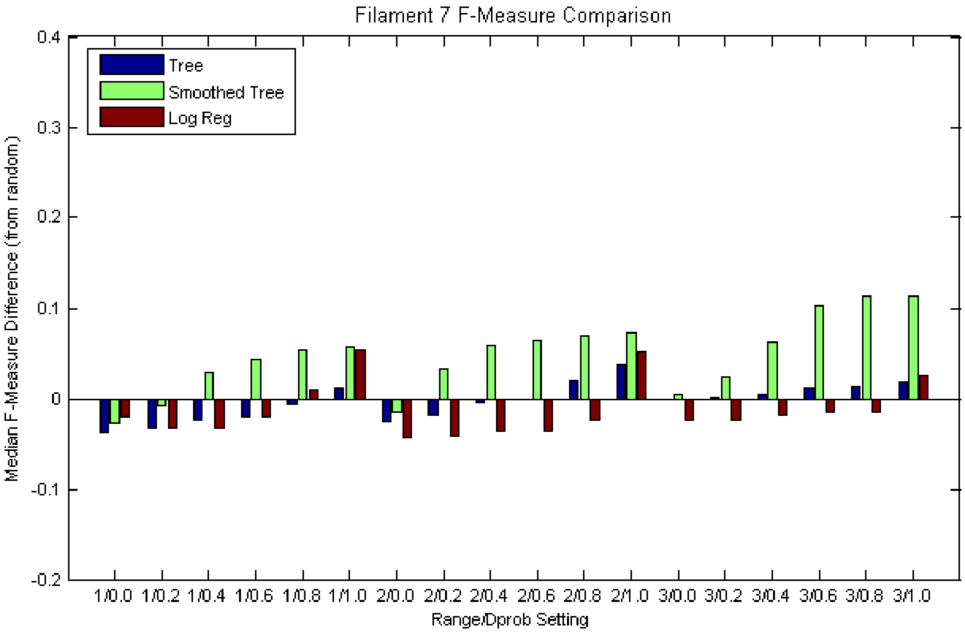


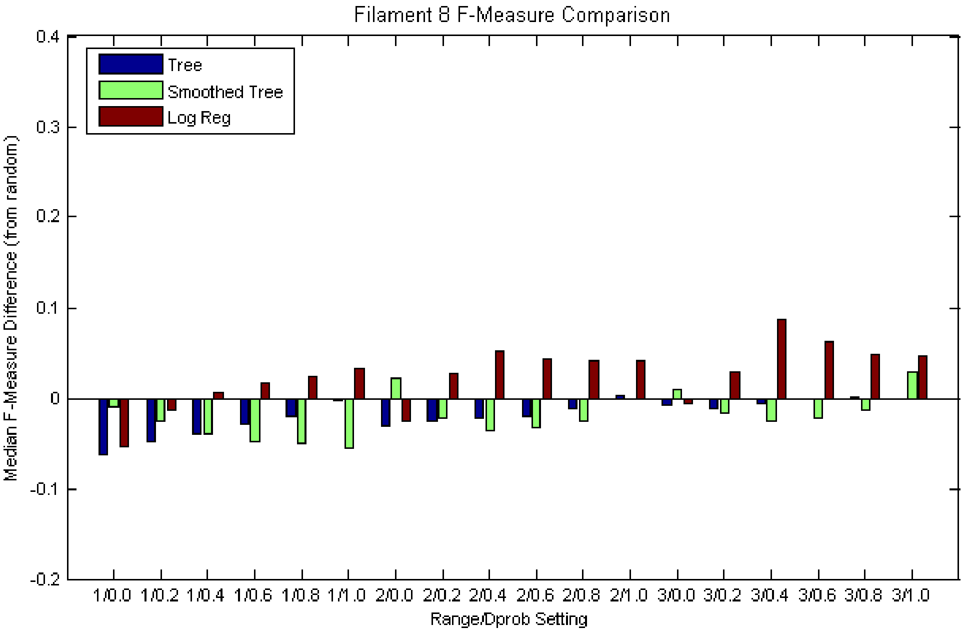
Filament 2 F-Measure Comparison

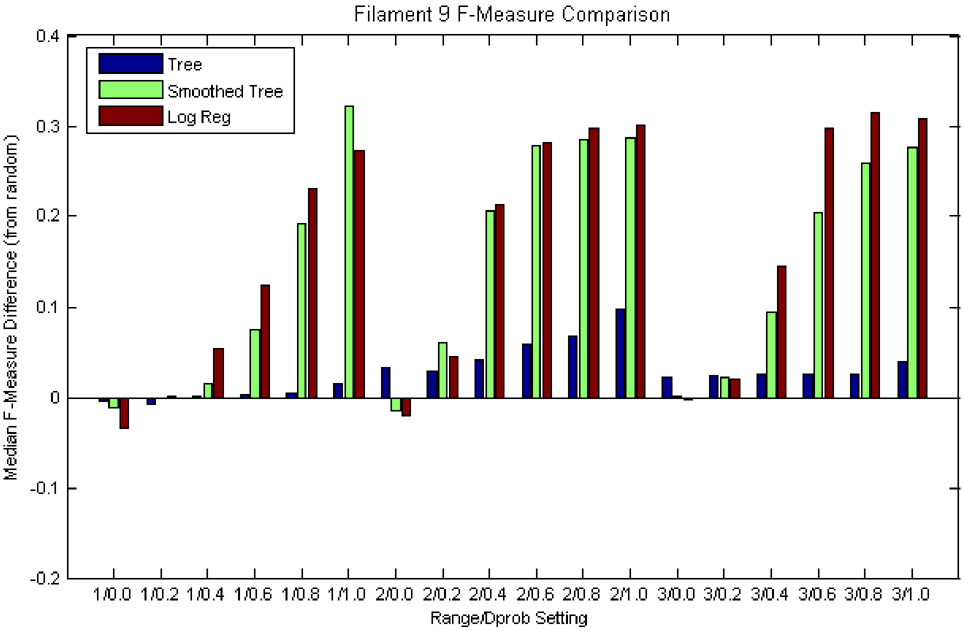

Filament 3 F-Measure Comparison
0.4
0.3
0.2
0.1
0
-0.1
-0.2
Median F-Measure Difference (from random)
Tree
Smoothed Tree
Log Reg
1/0.0
1/0.2
1/0.4
1/0.6
1/0.8
1/1.0
2/0.0
2/0.2
2/0.4
2/0.6
2/0.8
2/1.0
3/0.0
3/0.2
3/0.4
3/0.6
3/0.8
3/1.0
Range/Dprob Setting

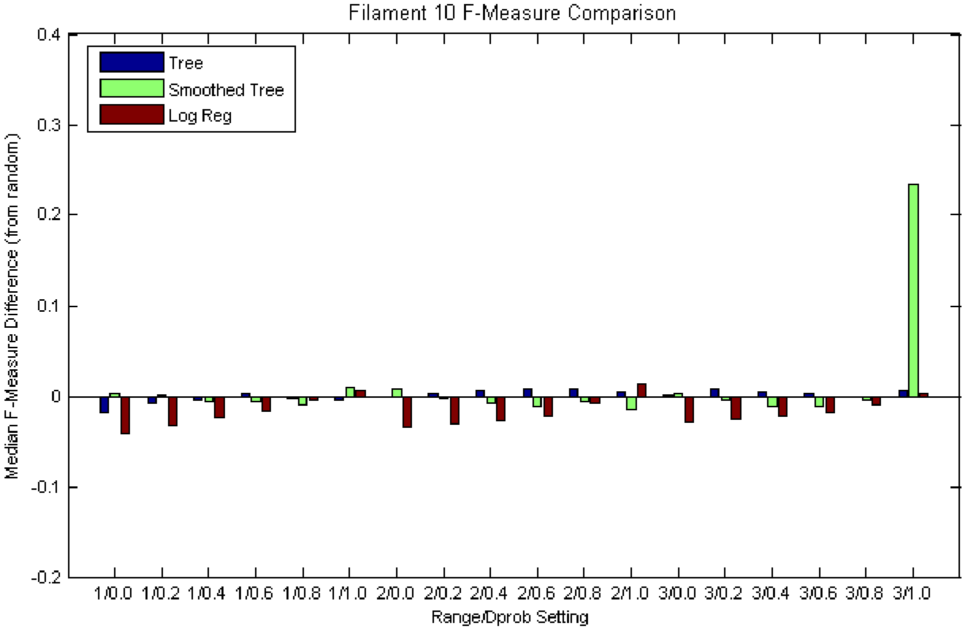

Filament 5 F-Measure Comparison
0.4
0.3
0.2
0.1
0
-0.1
-0.2
Median F-Measure Difference (from random)
Tree
Smoothed Tree
Log Reg
1/0.0
1/0.2
1/0.4
1/0.6
1/0.8
1/1.0
2/0.0
2/0.2
2/0.4
2/0.6
2/0.8
2/1.0
3/0.0
3/0.2
3/0.4
3/0.6
3/0.8
3/1.0
Range/Dprob Setting

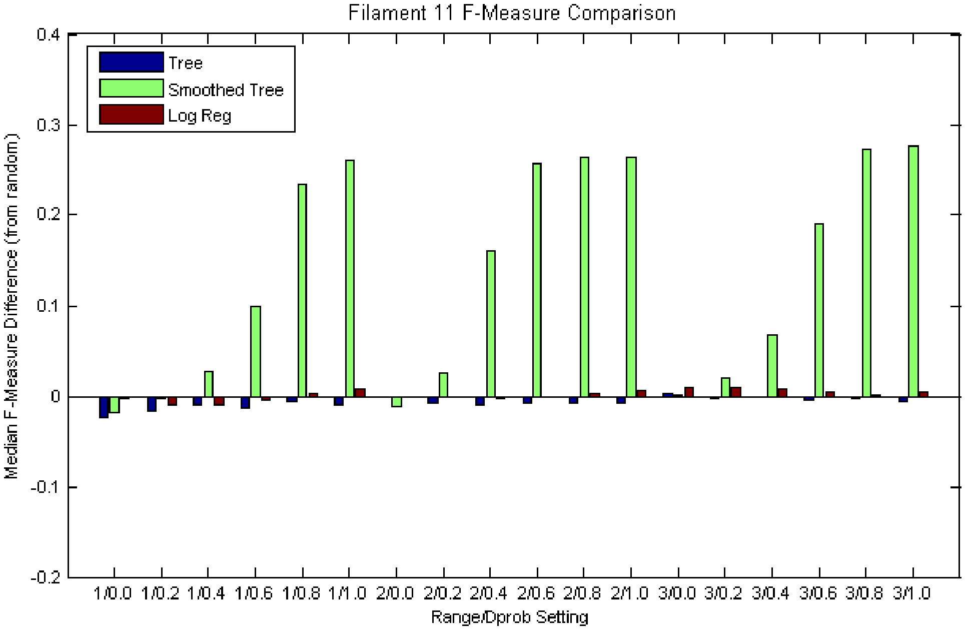

Filament 6 F-Measure Comparison
Tree
Smoothed Tree
Log Reg
Median F-Measure Difference (from random)
0.4
0.3
0.2
0.1
0
-0.1
-0.2
1/0.0
1/0.2
1/0.4
1/0.6
1/0.8
1/1.0
2/0.0
2/0.2
2/0.4
2/0.6
2/0.8
2/1.0
3/0.0
3/0.2
3/0.4
3/0.6
3/0.8
3/1.0
Range/Dprob Setting


Filament 7 F-Measure Comparison
Tree
Smoothed Tree
Log Reg
Median F-Measure Difference (from random)
0.4
0.3
0.2
0.1
0
-0.1
-0.2
1/0.0
1/0.2
1/0.4
1/0.6
1/0.8
1/1.0
2/0.0
2/0.2
2/0.4
2/0.6
2/0.8
2/1.0
3/0.0
3/0.2
3/0.4
3/0.6
3/0.8
3/1.0
Range/Dprob Setting


Filament 8 F-Measure Comparison
Tree
Smoothed Tree
Log Reg
Median F-Measure Difference (from random)
0.4
0.3
0.2
0.1
0
-0.1
-0.2
1/0.0
1/0.2
1/0.4
1/0.6
1/0.8
1/1.0
2/0.0
2/0.2
2/0.4
2/0.6
2/0.8
2/1.0
3/0.0
3/0.2
3/0.4
3/0.6
3/0.8
3/1.0
Range/Dprob Setting

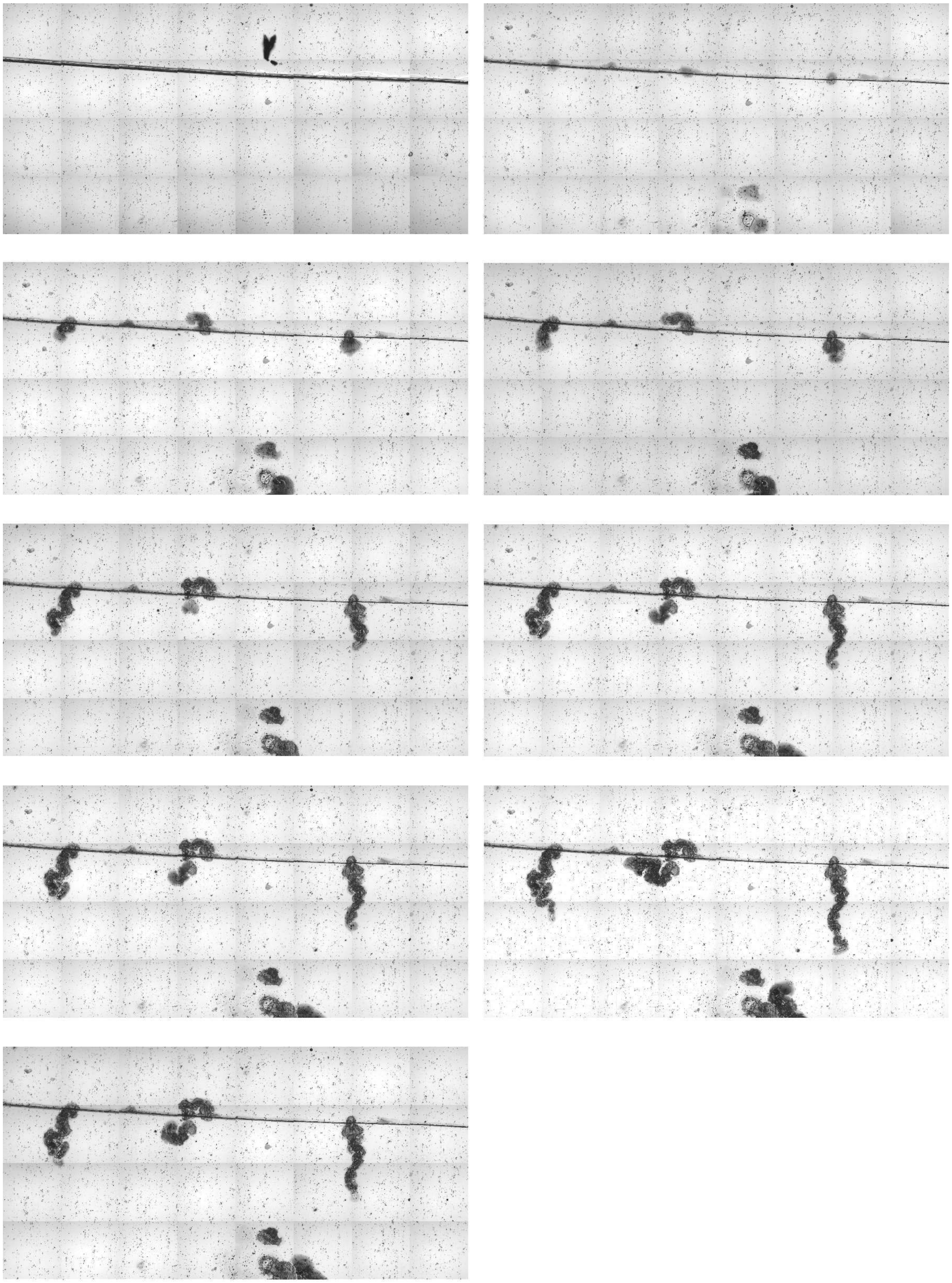

Filament 9 F-Measure Comparison
Tree
Smoothed Tree
Log Reg
Median F-Measure Difference (from random)
0.4
0.3
0.2
0.1
0
-0.1
-0.2
1/0.0
1/0.2
1/0.4
1/0.6
1/0.8
1/1.0
2/0.0
2/0.2
2/0.4
2/0.6
2/0.8
2/1.0
3/0.0
3/0.2
3/0.4
3/0.6
3/0.8
3/1.0
Range/Dprob Setting


Filament 10 F-Measure Comparison
Tree
Smoothed Tree
Log Reg
Median F-Measure Difference (from random)
0.4
0.3
0.2
0.1
0
-0.1
-0.2
1/0.0 1/0.2 1/0.4 1/0.6 1/0.8 1/1.0 2/0.0 2/0.2 2/0.4 2/0.6 2/0.8 2/1.0 3/0.0 3/0.2 3/0.4 3/0.6 3/0.8 3/1.0
Range/Dprob Setting

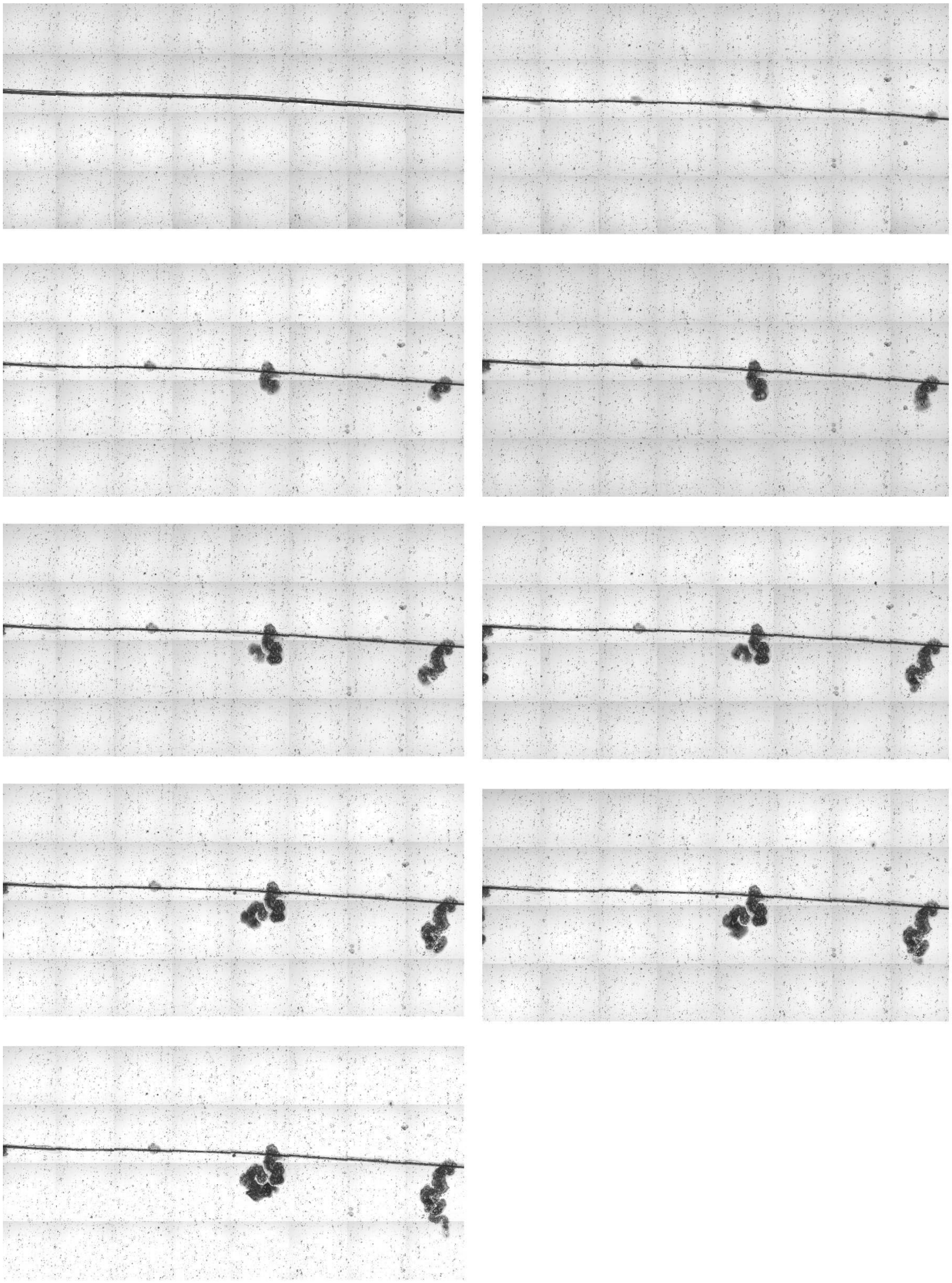

Filament 11 F-Measure Comparison
Tree
Smoothed Tree
Log Reg
Median F-Measure Difference (from random)
0.4
0.3
0.2
0.1
0
-0.1
-0.2
1/0.0 1/0.2 1/0.4 1/0.6 1/0.8 1/1.0 2/0.0 2/0.2 2/0.4 2/0.6 2/0.8 2/1.0 3/0.0 3/0.2 3/0.4 3/0.6 3/0.8 3/1.0
Range/Dprob Setting

Appendix H: Time lapse images of sample 1 areas

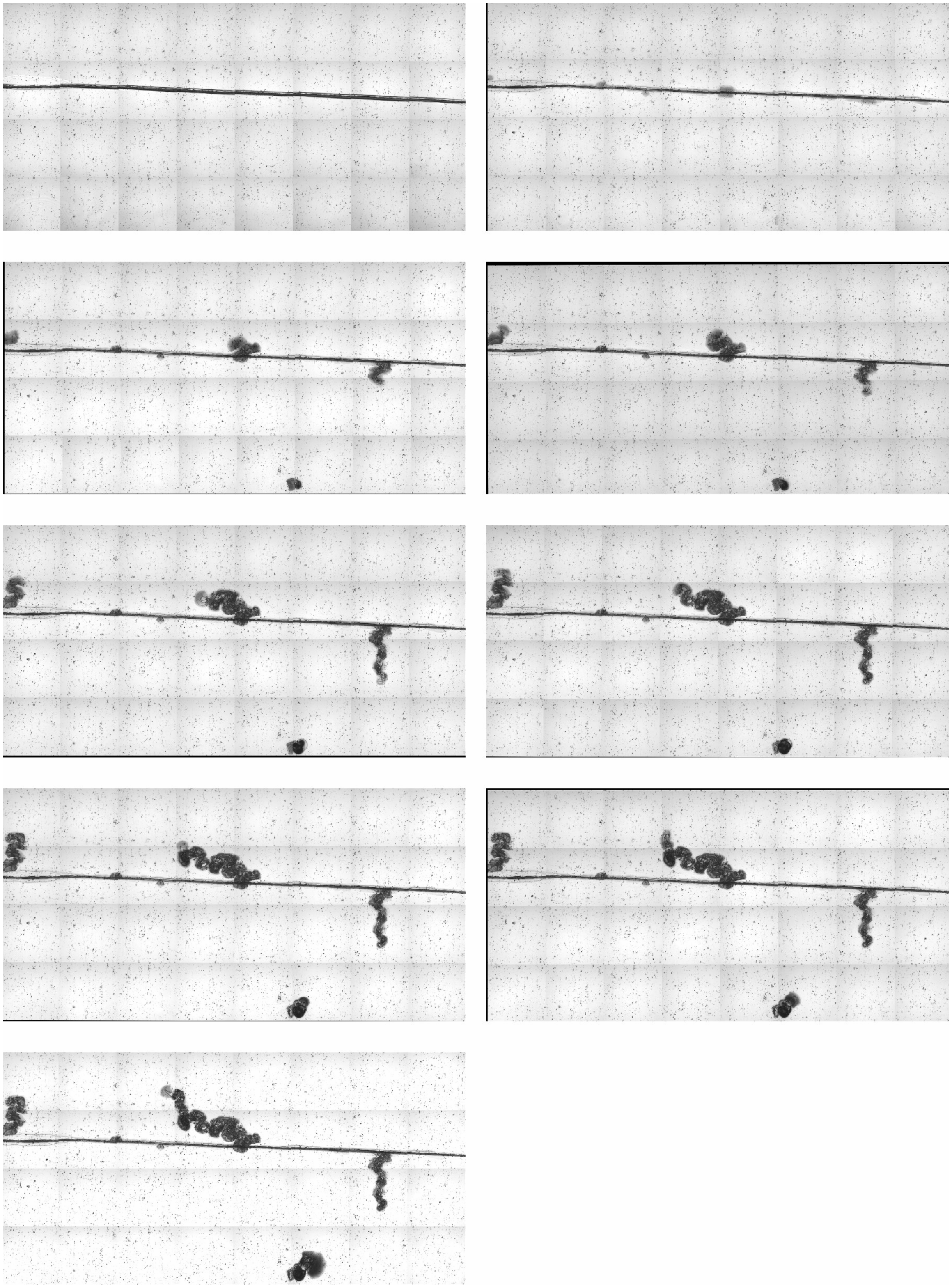
The following pages show the time lapsed images for Areas 3-11. The top left image is at time 0, and times {6, 21, 28, 44, 51, 67, 73, 89} continue from left to right and top to bottom.



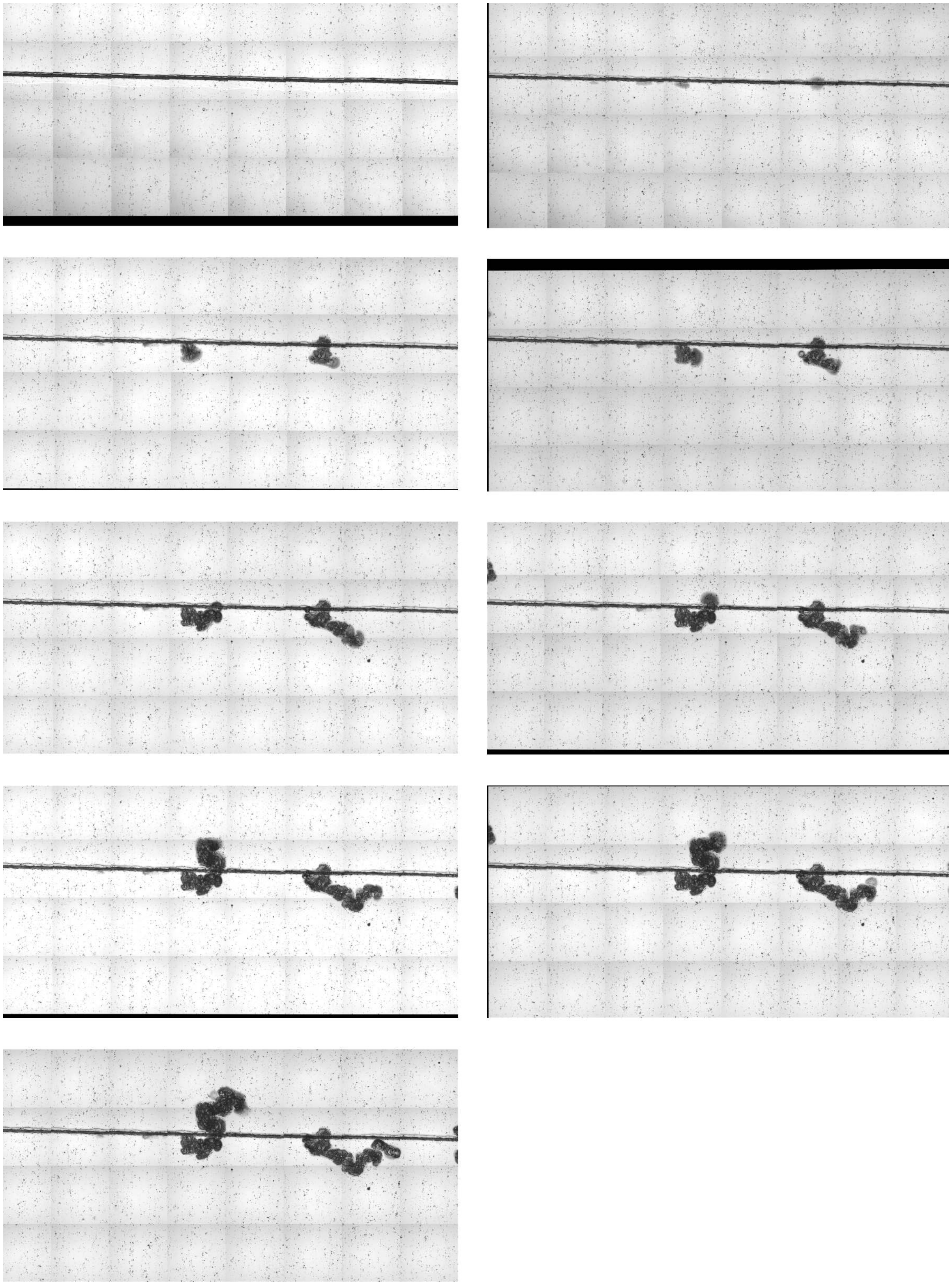
Area 3



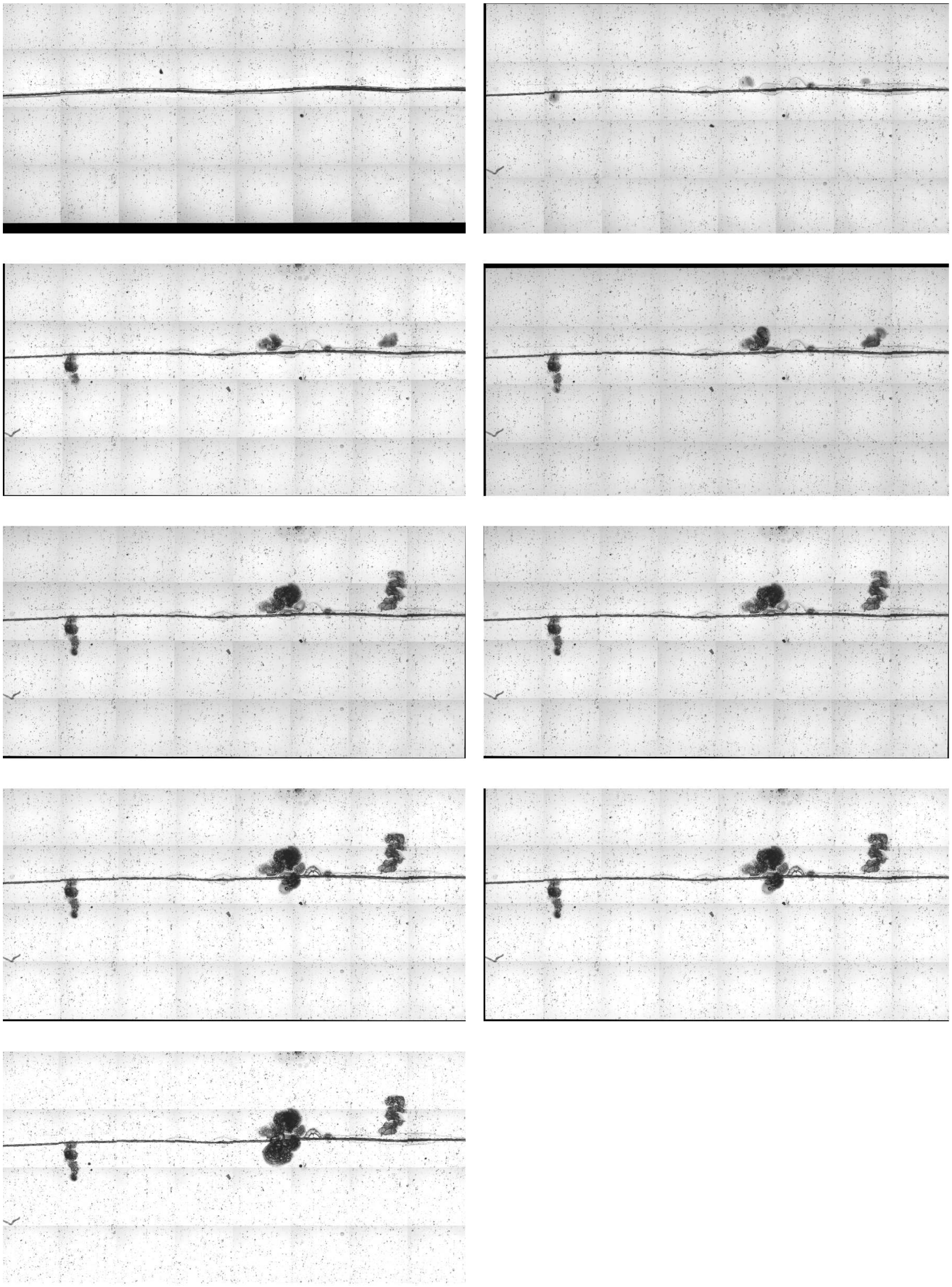
Area 4



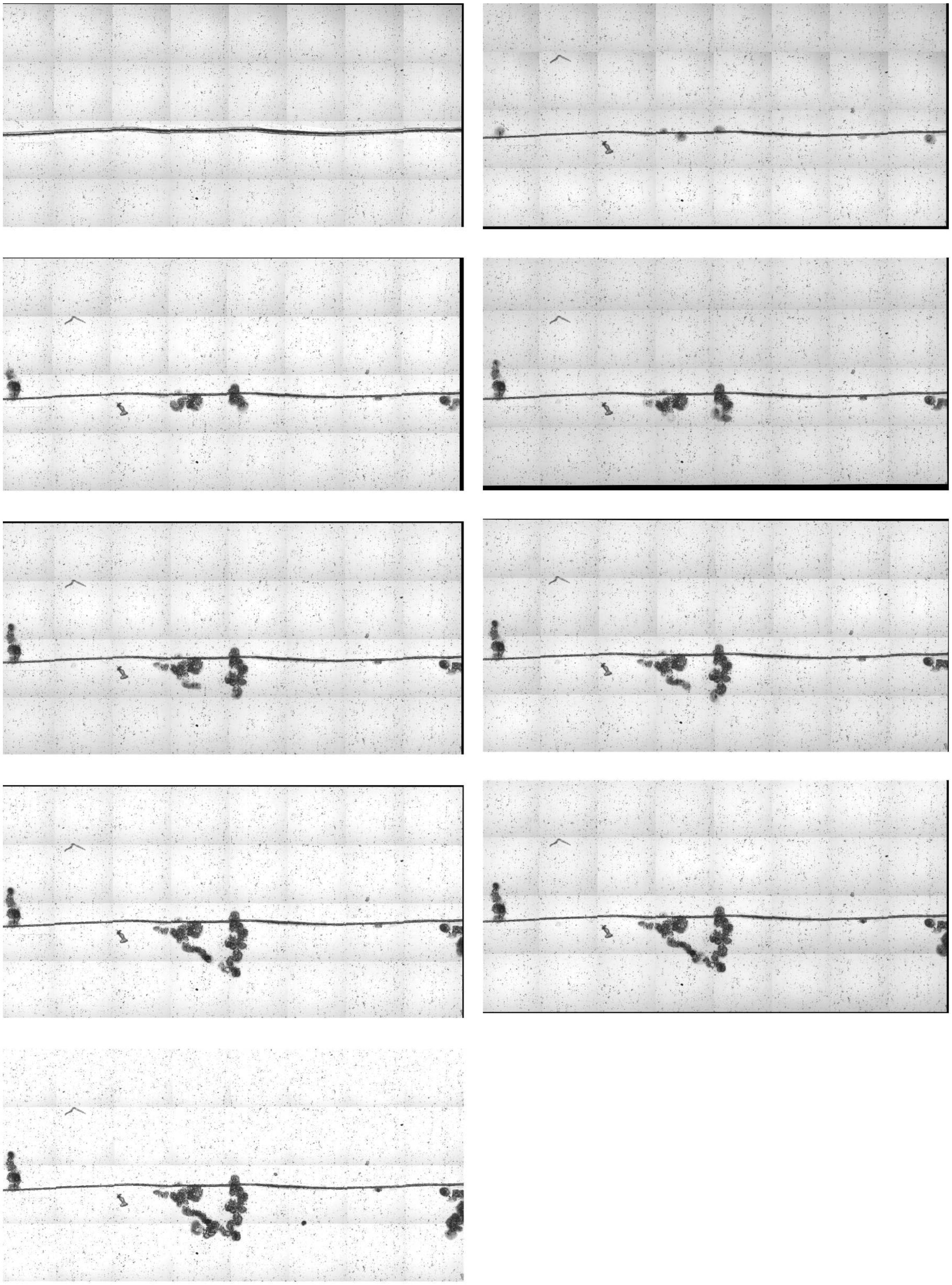
Area 5



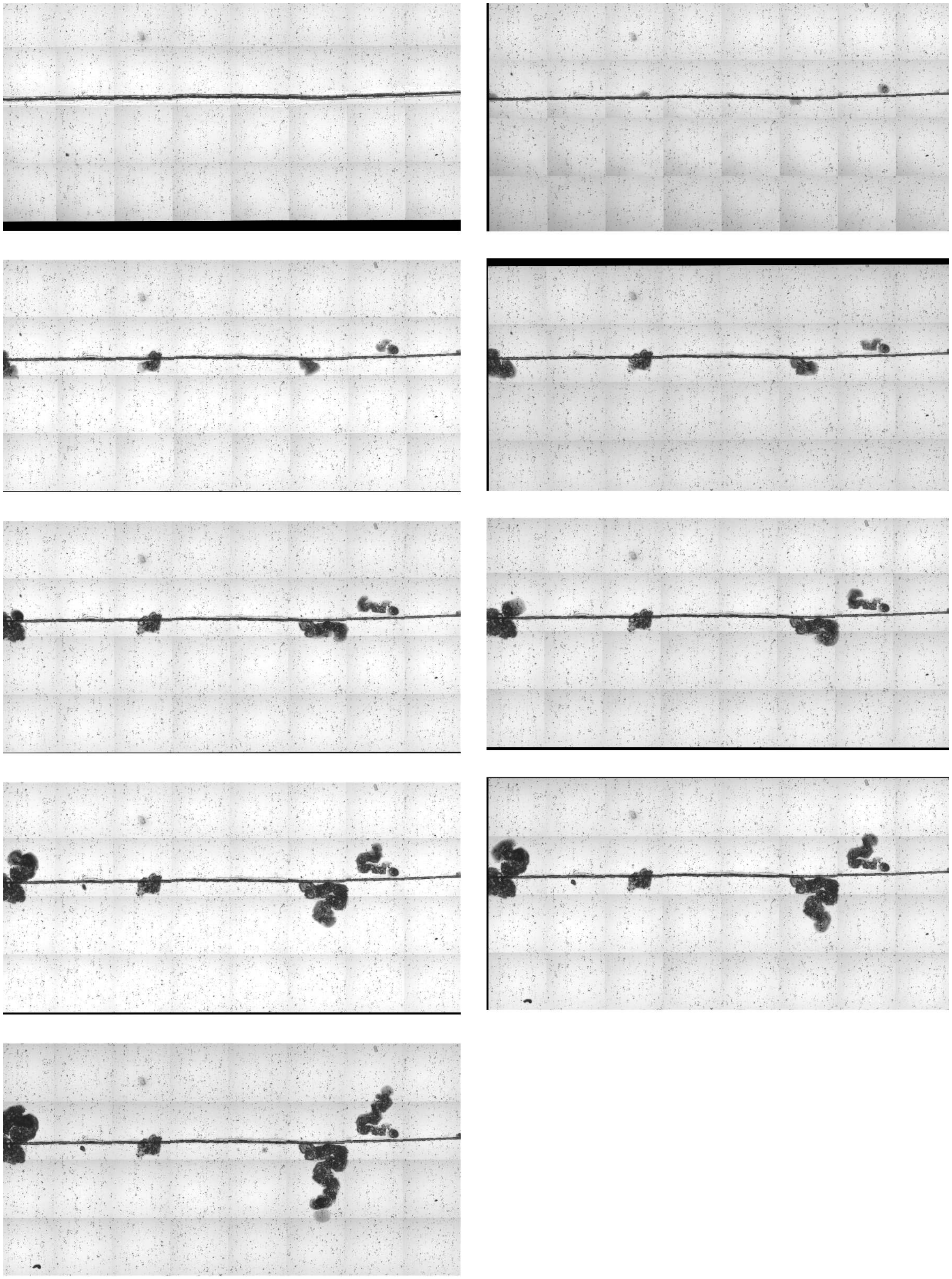
Area 6



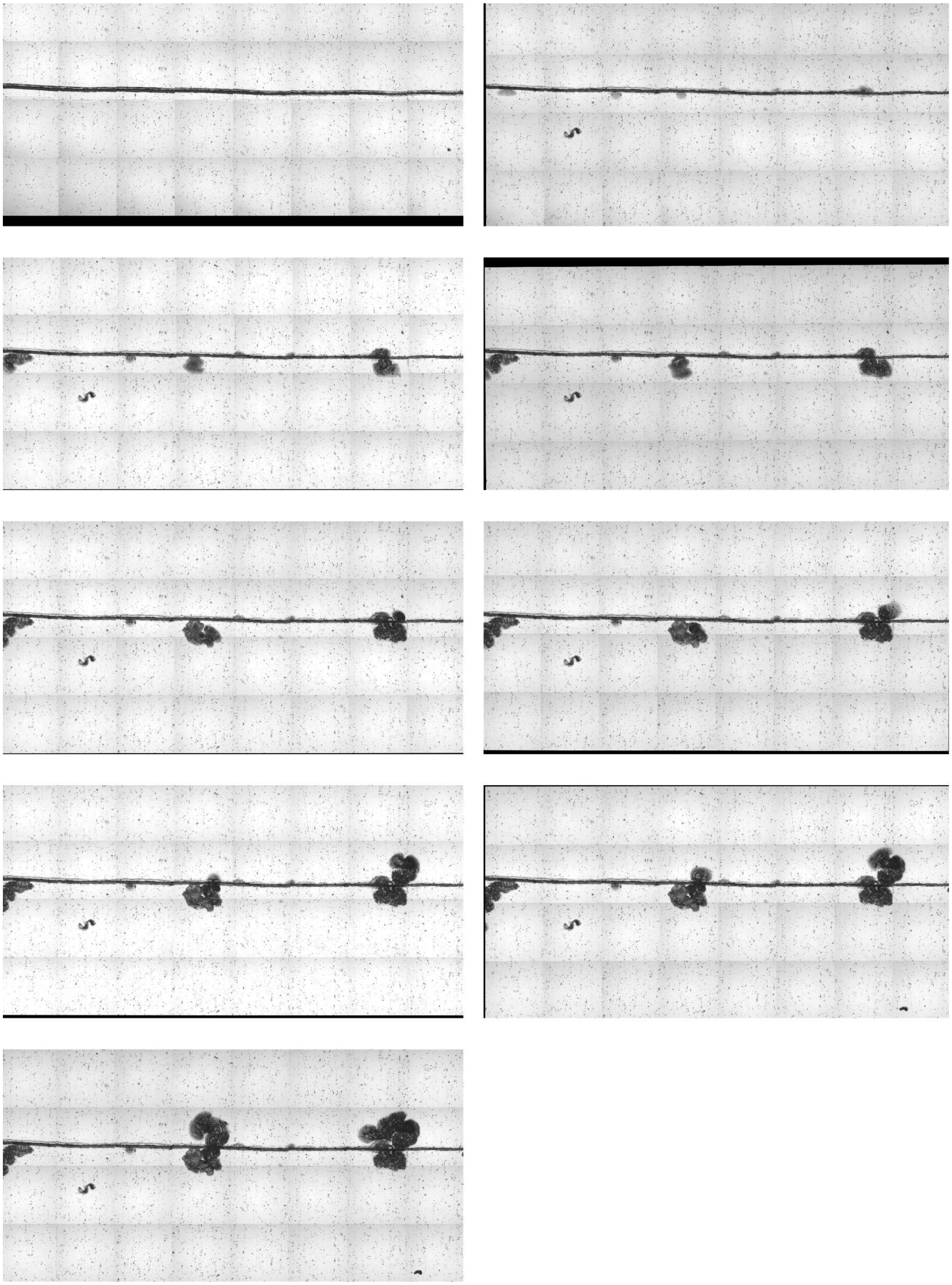
Area 7



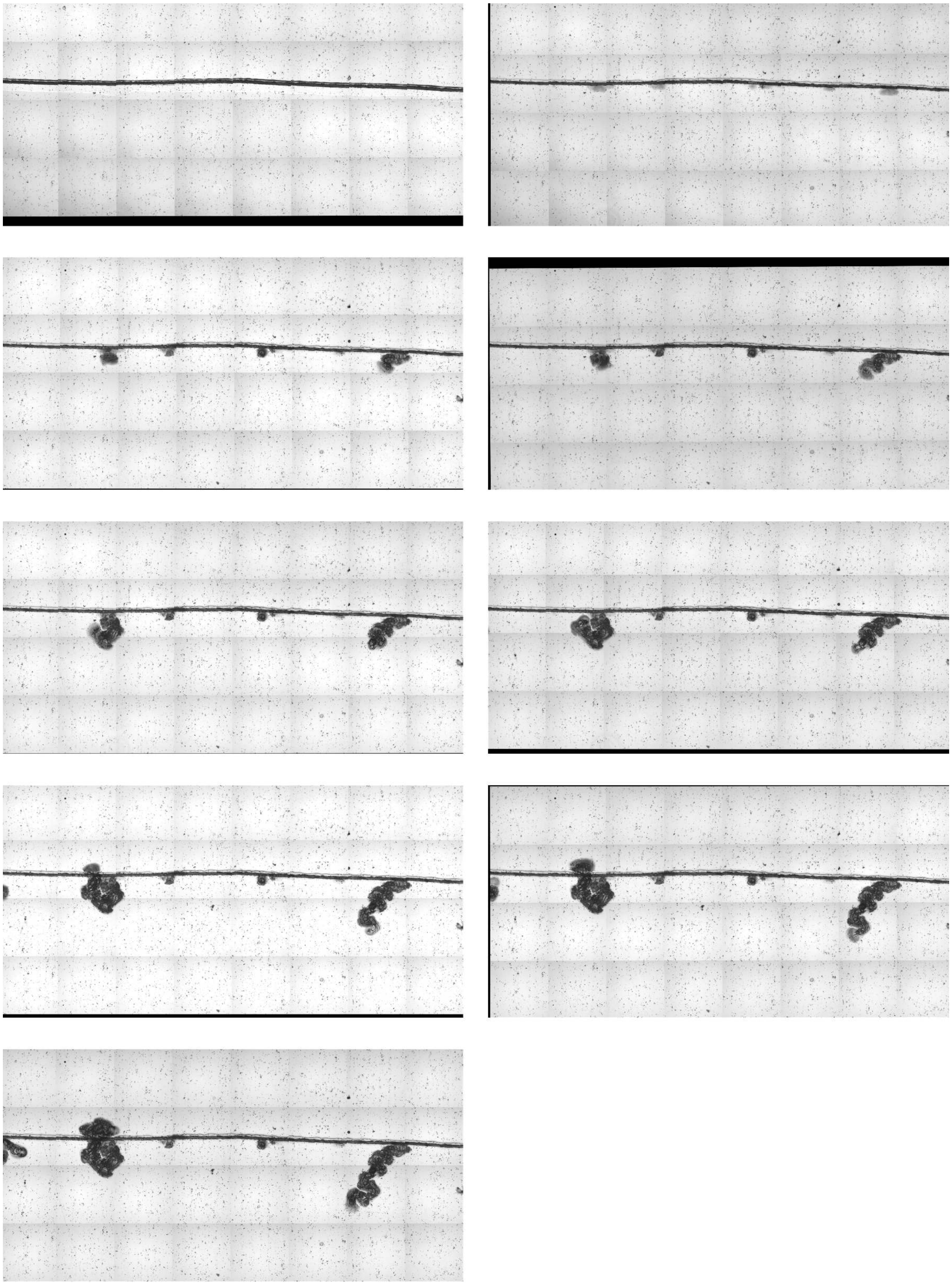
Area 8



Area 9



Area 10



Area 11

REFERENCES

AFRL (2004a). AFGROW History, Air Force Research Laboratory (AFRL). **2004**.

AFRL (2004b). "Air Force Corrosion Prevention and Control Office." Technology Horizons.

Allen, J. and K. Lu (2003). "Modeling and Prediction of Future Urban Growth in the Charleston Region of South Carolina: a GIS-based Integrated Approach." Ecology and Society **8**(2).

Ariaratnam, S. T., El-Assaly, et al. (2001). "Assessment of Infrastructure Inspection Needs Using Logistic Models." Journal of Infrastructure Systems, ASCE **7**(4): 160-165.

Aristoff, J. M. (2003). "The Diffusion-Limited Aggregate." MIT Undergraduate Journal of Mathematics: 1-10.

Aziz, M. (1956). "Application of the Statistical Theory of Extreme Values to the Analysis of Maximum Pit Depth Data for Aluminum." Corrosion **12**: 495.

Bell, J. F. (1996). "Application of classification trees to the habitat preference of upland birds." Journal of Applied Statistics **23**(2-3): 349-359.

Berjak, S. G. and J. W. Hearne (2002). "An improved cellular automaton model for simulating fire in a spatially heterogeneous Savanna system." Ecological Modeling **148**: 133-151.

Bonnie, G. E. (1987). "Logistic Regression for Dependent Binary Observations." Biometrics **43**(4): 951-973.

Booth, F. F. and G. E. G. Tucker (1965). "Statistical Distribution of Stree Corrosion Endurance." Corrosion **21**: 173.

Breiman, L. (1996a). "Bagging Predictors." Machine Learning **24**(2): 123-140.

Breiman, L. (1996b). Bias, Variance, and Arcing Classifiers, University of California.

Breiman, L., J. H. Friedman, et al. (1984). Classification and Regression Trees. Blemont, CA, Wadsworth International Group.

Brence, J. R. and D. E. Brown (2002a). Complexity Challenges to the Discovery of Corrosion Relationships in Eddy Current Non-Destructive Test Data. Mathematical Challenges in Scientific Data Mining, Institute for Pure and Applied Mathematics, University of California, Los Angeles.

Brence, J. R. and D. E. Brown (2002b). "Data Mining Corrosion From Eddy Current Non-Destructive Tests." Computers & Industrial Engineering **43**: 821-840.

Brown, M. P. S., W. N. Grundy, et al. (1999). Support Vector Machine Classification of Microarray Gene Expression Data. Proceedings of the National Academic Society, CA.

Budiansky, N. D. and J. R. Scully (2002). Origins of Persistent Interaction Among Localized Corrosion Sites. Electrochemical Society Proceedings.

Buxton, D. C. and P. A. Scarf (1994). Life Prediction of Corrodible Structures. Houston, TX, NACE.

Cai, J., R. A. Cottis, et al. (1999). "Phenomenological modeling of atmospheric corrosion using an artificial neural network." Corrosion Science **41**: 2001-2030.

Clark, P. J. and F. C. Evans (1954). "Distance to Nearest Neighbor as a Measure of Spatial Relationships in Populations." Ecology **35**: 445-453.

Cliff, A. D., P. Haggett, et al. (1975). Elements of Spatial Structure: A Quantitative Approach. New York, NY, Cambridge University Press.

Cooley, C. and S. MacEachern (1998). "Classification via kernel product estimators." Biometrika **85**(4): 823-833.

Cressie, N. (1993). Statistics for Spatial Data. New York, NY, John Wiley and Sons.

Cressie, N. and F. L. Hulting (1992). "A Spatial Statistical Analysis of Tumor Growth." Journal of the American Statistical Association **87**(418): 272-283.

Cressie, N. and G. M. Laslett (1987). "Random Set Theory and Problems of Modeling." SIAM Review **29**(4): 557-574.

Diggle, P. J. (1983). Statistical Analysis of Spatial Point Patterns. New York, NY, Academic Press Inc.

Draper, N. R. and H. Smith (1981). Applied Regression Analysis. New York, NY, John Wiley & Sons.

Edelman, G. (1988). Topobiology: An Introduction to Molecular Embryology. New York, NY, Basic Books.

Eden, M. (1958). A Probabilistic Model for Morphogenesis. Symposium for Information Theory in Biology, New York, NY, Pergamon Press.

EFUNDA (2005). Aluminum Alloys, Engineering Fundamentals. **2004**.

Engelhardt, G. R., D. D. Macdonald, et al. (1999). "Development of fast algorithms for estimating stress corrosion crack growth rate." Corrosion Science **41**: 2267.

Fiksel, T. (1984). "Simple Spatial-Temporal Models for Sequences of Geological Events." Elektronische Informationsverarbeitung und Kybernetik **20**: 480-487.

Fisher, R. A. (1922). "The Accuracy of the Plating Method of Estimating the Density of Bacterial Population." Annals of Applied Biology **9**: 325-359.

Garamone, J. (2001). As Equipment Ages, Readiness Suffers. American Forces Information Service (Defense Link News).

Gordon, A. D. (1999). Classification. Boca Raton, Fl, Chapman & Hall/CRC.

Greene, N. (1989). "Voxel Space Automata: Modeling with Stochastic Growth Processes in Voxel Space." Computer Graphics **23**(3): 175-184.

Griffeath, D. (1993). "Frank Spitzer's Pioneering Work on Interacting Particle Systems." The Annals of Probability **21**(2): 608-621.

Halsey, T. C. (2000). Diffusion-Limited Aggregation: A Model for Pattern Formation, American Institute of Physics.

Hastie, T., R. Tibshirani, et al. (2001). The Elements of Statistical Learning: Data Mining, Inference, and Prediction. New York, NY, Springer.

Hermann, H. J. (1986). "Geometrical Cluster Growth Models and Kinetic Gelation." Physics Reports **136**(3): 153-227.

Hermann, P. U. a. H. (1996). "Computer simulation of crystallization kinetics with non-Poisson distributed nuclei." Modeling and Simulation in Materials Science and Engineering **4**: 203-214.

Hosmer, D. W. and S. Lemeshow (2000). Applied logistic regression. New York, NY, Wiley.

Hug, D., G. Last, et al. (2002). "Generalized Contact Distributions of Inhomogeneous Boolean Models." Advanced Applied Probability **34**: 21-47.

Ivanenko, Y. V., N. I. Lebovka, et al. (1999). "Eden growth model for aggregation of charged particles." The European Physical Journal B **11**: 469-480.

Karafyllidus, I. and A. Thanailakis (1997). "A model for predicting forest fire spreading using cellular automata." Ecological Modeling **99**: 87-97.

Kelly, R. G., J. R. Scully, et al. (2003). Electrochemical Techniques in Corrosion Science and Engineering. New York, NY, Marcel Dekker Inc.

Kingston (1999). Corrosion Theory - Pitting Corrosion, Corrosion Doctors. **2004**.

Koch, G. H., L. Yu, et al. (1998). "Mathematical Model to Predict Fatigue Crack Initiation in Corroded Lap Joints."

Leifer, J., P. E. Zapp, et al. (1999). "Predictive models for determination of pitting corrosion versus inhibitor concentrations and temperature for radioactive sludge in carbon steel waste tanks." Corrosion **55**(1).

Li, X. and W. Magill (2001). "Modeling fire spread under environmental influence using a cellular automaton approach." Complexity International **8**: 1-14.

Liggett, T. M. (1985). Interacting Particle Systems. New York, NY, Springer-Verlag Inc.

Liggett, T. M. (1997). "Stochastic Models of Interacting Systems." The Annals of Probability **25**(1): 1-29.

Liu, H. (1999). Space-Time Prediction: A New Model, University of Virginia.

Liu, H. and D. E. Brown (2003). "Criminal incident prediction using a point-pattern based density model." International Journal of Forecasting **19**(4): 603-622.

Liu, P.-S. and Y. H. Chou (1997). A Grid Automation of Wildfire Growth Simulation. Environmental Systems Research Institute User Conference.

Lunt, T. T., S. T. Pride, et al. (1997). "Cooperative Stochastic Behavior in Localized Corrosion, II. Experiments." Journal of the Electrochemical Society **144**(5): 1620-1629.

Macdonald, D. D. and G. Engelhardt (2003). Theory for the Deterministic Prediction of Pitting Corrosion Damage on Aluminum. Electrochemical Society Proceedings.

Mase, J. (1982). "Properties of Fourth Order Strong Mixing Rates and Its Application to Random Set Theory." Journal of Multivariate Analysis **12**: 549-561.

Matheron, G. (1975). Random Sets and Integral Geometry. New York, NY, John Wiley and Sons.

McCullagh, P. and J. A. Nelder (1989). Generalized Linear Models. London, Chapman and Hall.

Mears, P. B. and U. R. Evans (1937). Industrial Engineering Chemistry **29**: 1087.

Menard, S. (2001). Applied Logistic Regression Analysis. Thousand Oaks, CA, Sage Publications.

Molchanov, I. (1997). Statistics of the Boolean Model for Practitioners and Mathematicians. New York, NY, John Wiley and Sons.

Molchanov, I. and V. Scherbakov (2003). "Coverage of the Whole Space." Advanced Applied Probability **35**: 898-912.

NAP (1997). Aging of U.S. Air Force Aircraft, The National Academy Press.

Neary, D. (2000). Fractal Methods in Image Analysis and Coding, Dublin City University.

Nesic, S., M. Nordsveen, et al. (2001). "Probabilistic modeling of CO_2 corrosion laboratory data using neural networks." Corrosion Science **43**: 1373-1392.

Neyman, J. and E. Scott (1958). "Statistical Approach to Problems of Cosmology." Journal of the Royal Statistical Society **20**: 1-43.

Otsu, N. (1979). "A Thresholding Selection Method from Gray Level Histogram." IEEE Transaction on Systems, Man, and Cybernetics **9**: 62-66.

Pal, N. R. and S. K. Pal (1993). "A review of image segmentation techniques." Pattern Recognition **26**: 1277-1294.

Pegden, D. C., R. E. Shannon, et al. (1995). Introduction to Simulation Using SIMAN. Blacklick, OH, McGraw-Hill.

Pfeifer, P. E. and S. J. Deutsch (1980). "A Three-Stage Iterative Procedure for Space-Time Modeling." Technometrics **22**(1): 35-47.

Pintos, S., N. V. Queipo, et al. (2000). "Artificial neural network modeling of atmospheric corrosion in the MICAT project." Corrosion Science **42**: 35-52.

Preisler, H. K., D. R. Brillinger, et al. (2004). "Probability based models for estimation of wildfire risk." International Journal of Wildland Fire **13**: 133-142.

Rikvold, P. A. (1982). "Simulations of a stochastic model for cluster growth on a square lattice." Physical Review A **26**(1): 647-650.

Ripley, B. D. (1977). "Modeling Spatial Patterns." Journal of the Royal Statistical Society **39**: 172-192.

Salvago, G., L. Magagnin, et al. (2002). "Unified Approach to Localized and General Corrosion of Stainless Steels." Electrochimica Acta **47**: 1787-1973.

Scott, D. W. (1992). Multivariate Density Estimation. New York, NY, John Wiley & Sons, Inc.

Serra, J. (1980). "The Boolean Model and Random Sets." Computer Graphics and Image Processing **12**: 99-126.

Serra, J. (1982). Image Analysis and Mathematical Morphology. New York, NY, Academic Press Inc.

Shao, J. (2003). Mathematical Statistics. New York, NY, Springer-Verlag.

Shibata, T. (2000). Uhlig's Corrosion Handbook. New York, NY, John Wiley and Sons.

Shibata, T. and T. Takeyama (1977). Boshaku-Gijutsu **26**: 25.

Shumway, R. H. and D. S. Stoffer (2005). Time Series Analysis and Its Applications. New York, NY, Springer Science.

Smith, J. A. and A. F. Karr (1985). "Statistical Inference for Point Process Models of Rainfall." Water Resources Research **21**: 73-79.

Smith, M. A. (2005). Choice Modeling of Bombing Attach Site Selection, University of Virginia.

Sullivan, A. L. and I. K. Knight (2004). A hybrid cellular automata/semi-physical model of fire growth. The 7th Asia Pacific Conference on Complex Systems, Cairns, Australia.

Thompson, S. and A. Rosenfeld (1995). "Isotropic Growth on a Grid." Pattern Recognition **28**(2).

Thompson, S. F. (1994). Growth Models for Shapes, University of Maryland.

Thompson, S. F. and A. Rosenfeld (1999). "Parallel array grammars as models for the growth of planar patterns." Pattern Recognition **32**: 255-267.

Virmani, Y. P. (2002). Corrosion Costs and Preventative Strategies in the United States. Dublin, OH, CC Technologies Laboratories Inc.

Wolfram (2004). Digital Image Processing Documentation, Wolfram Research.

Zhou, X., X. Wang, et al. (2005). "Gene Selection Using Logistic Regressions Based on AIC, BIC, and MDL criteria." New Mathematics and Natural Computation **1**(1): 129-145.